International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Approaches to handle Data Imbalance Problem in Predictive Machine Learning Models: A Comprehensive Review

Govind M. Poddar[1], Rajendra V. Patill [2], Dr. Satish Kumar N[3]

Submitted: 06/02/2024 **Revised**: 14/03/2024 **Accepted**: 20/03/2024

Abstract: The business organizations ability to grow and flourish mostly relies on how successfully it understands and utilizes the data it has collected; data has become more vital in today's society. Every company or organization at the present time accumulates massive volumes of data across a range of areas, such as finance, trade, business, and healthcare. Medical data may be provided by clinics, doctors, healthcare providers, and insurance establishments. Upon locating the necessary medical datasets, the next phases would be to investigate and utilize appropriate modeling algorithms to mine substantial information for probable prediction. Biased data is significant challenge in machine learning where the distribution of data elements in a dataset is uneven, with one class considerably outnumbering the others. This occurrence leads to biased models and reduced performance that affects quality and reliability of machine learning algorithms. This paper presents detailed review on reasons for imbalanced data, its impact, algorithmic procedures to handle unevenly distributed data. We explore various techniques, algorithms to address problem, advantages, demerits and evaluation metrics to assess performance of procedures for handling imbalanced datasets.

Keywords: *Machine Learning, Imbalanced data, Sampling, Data Preprocessing, Data Mining, Class imbalance*

1. Introduction

Machine learning techniques has transformed the world of artificial intelligence and information analysis, enabling machines to learn and make forecasts or judgments without explicit programming [1, 2]. These algorithms are at the core of numerous applications, from recommendation systems to autonomous vehicles and healthcare diagnostics. Machine learning has appeared as a transformative force in the domain of healthcare, proposing inventive solutions to long-lasting challenges and revolutionizing the way healthcare professionals diagnose, treat, and manage patients [3]. This technology has ushered in a new era of personalized medicine, data-driven decision-making, and improved patient outcomes.

Improving diagnosis accuracy is one of machine learning's primary applications in the medical field. Large volumes of patient data, such as genetic data, clinical records, and medical imaging, may be analysed by machine learning algorithms to find patterns and abnormalities that human physicians would miss [3, 4]. This capacity is especially useful in domains such as radiology, where machine learning may help with the early diagnosis of illnesses like cancer, heart problems, and neurological disorders, resulting in earlier therapies and better predictions..

Moreover, machine learning algorithms can aid in predicting patient outcomes and disease progression. By analysing historical patient data, including demographic information, medical history, and treatment outcomes, these algorithms can deliver healthcare practitioners with valued insights into a patient's risk factors and the likely trajectory of their condition [2, 3, 4]. This predictive power enables proactive interventions and personalized treatment plans, ultimately improving patient care.

Another pivotal role of machine learning in healthcare is the automation of administrative tasks and workflow optimization. Healthcare facilities deal with a vast amount of paperwork, scheduling, and billing processes, which can be laborious and error-prone [2-5]. Machine learning-driven systems can streamline these operations, sinking organizational loads and permitting healthcare specialists to concentrate more on patient care.

Furthermore, the research and discovery of new drugs is greatly aided by machine learning [5]. A new medicine must frequently be developed through an expensive and time-consuming procedure that costs billions of dollars and takes years to bring to market. Large-scale information may be analysed by machine learning to find possible medication candidates, forecast their effectiveness, and model how they will interact with the body. This lowers expenses, quickens the process of discovering new drugs, and may result in the creation of more potent therapies for a range of illnesses.

The chief objective of this paper is to outline imbalanced dataset problem in predictive models. First, this work will identify importance of predictive models in different application domains. In Section 3 to 5, we present detailed description of difference between balanced data and biased

[1] *Research Scholar, SunRise University, Alwar, Rajasthan, India Associate Professor, Gangamai College of Engineering, Nagaon, Dhule (M.S.), Dhule, ORCID ID : 0009-0007-7185-0001*
[2] *Assistant Professor, SSVPS BSD College of Engg, Dhule (M.S.), India ORCID ID : 0009-0000-1105-0423*
[3] *Research Supervisor, Sunrise University, Alwar, Rajastha, India*
** Corresponding Author Email: patilrajendra.v@gmail.com*

data with reasons for class imbalance with its impact on performance of predictive models. Section 6 presents detailed description of algorithms proposed in literature to tackle with imbalance datasets. In Section 6, this work compares approaches for handling imbalanced datasets with advantages and limitations. In order to select best suited method to solve data imbalance problem, evaluations metrics used to assess performance of techniques are described in section 7. Finally, Section 8 will provide practical recommendations and future directions for effectively dealing with imbalanced data in real-world scenarios.

2. Significance of Predictive models

Predictive models play a pivotal role in various fields and industries, owing to their ability to anticipate future outcomes based on historical data and patterns. These models have revolutionized decision-making processes, enabling organizations to make informed choices, allocate resources efficiently, and mitigate risks effectively. Whether in finance, healthcare, marketing, or even weather forecasting, the importance of predictive models cannot be overstated.

In finance, predictive models [6] are indispensable tools for assessing investment risks and optimizing portfolios. By analyzing historical market data, these models can forecast asset prices, identify trends, and provide valuable insights into market behavior. Investors and financial institutions rely on these predictions to make strategic decisions, maximize returns, and minimize losses. Moreover, predictive models are essential for credit scoring, helping lenders assess borrowers' creditworthiness and reduce default rates.

In healthcare, predictive models [2-5] have emerged as a game-changer in patient care and resource allocation. Healthcare providers use these models to predict disease outbreaks, patient readmission rates, and even individual patient outcomes. By analyzing patient data and medical records, healthcare professionals can identify high-risk patients and tailor treatment plans accordingly. Predictive models also assist in optimizing hospital operations, such as staff scheduling and resource allocation, leading to improved patient care and cost savings.

The marketing industry [7, 37] heavily relies on predictive models to target customers effectively and optimize advertising campaigns. These models analyze customer behavior, purchase history, and demographics to predict future buying patterns and preferences. Marketers can then personalize their marketing strategies, recommend products, and send targeted promotions to increase conversion rates and customer satisfaction. Predictive models also help in customer churn prediction, enabling businesses to proactively retain customers and reduce attrition.

In manufacturing and supply chain management [8], predictive models are instrumental in optimizing production processes and reducing operational costs. Manufacturers use these models to forecast demand, manage inventory efficiently, and minimize production downtime. By analyzing historical production data and external factors like market trends and supplier performance, businesses can make data-driven decisions to improve production efficiency and meet customer demands on time.

Predictive models have a significant impact on environmental science and conservation efforts. Climate scientists use these models to predict climate change patterns, assess the impact of human activities on the environment, and develop strategies for mitigation and adaptation. Conservationists use predictive models to track the movement and behavior of endangered species, helping protect and preserve biodiversity.

In the field of transportation [1-9], predictive models are crucial for optimizing routes, reducing traffic congestion, and enhancing safety. Transportation companies use these models to forecast demand for services, improve vehicle maintenance schedules, and plan efficient routes. City planners and traffic management authorities rely on predictive models to anticipate traffic patterns and make informed decisions to alleviate congestion and enhance public transportation systems.

Weather forecasting heavily depends on predictive models to provide accurate and timely weather predictions [1-9]. Meteorologists analyze vast amounts of atmospheric data, historical weather patterns, and satellite imagery to develop sophisticated models that can forecast weather conditions, severe storms, and natural disasters. These predictions are vital for public safety, agriculture, and disaster preparedness.

Predictive models are also indispensable in the field of education, where they help identify students at risk of academic failure. By analyzing students' academic performance, attendance records, and socio-economic factors, educators can intervene early to provide additional support and resources, ultimately improving student outcomes.

Predictive models have become essential tools across a wide range of industries and applications [1-9]. They enable organizations to harness the power of data and make more informed decisions, leading to improved efficiency, cost savings, and better outcomes. As technology continues to advance and data becomes more abundant, the importance of predictive models in shaping our future cannot be overstated. These models will continue to play a central role in helping us navigate complex challenges and unlock new opportunities in an increasingly data-driven world.

3. Even Distribution Versus Uneven Distribution

If the positive and negative values in a dataset are nearly equal, it is said to be balanced [10 -12]. On the other hand, unbalanced datasets are unique situations in which there is an uneven distribution of values across the classes, giving rise to two separate classes: the majority, or negative class, and the minority, or positive class. Both balanced and unbalanced datasets are shown in Figure 1, with the positive values displayed in orange and the negative values in blue.

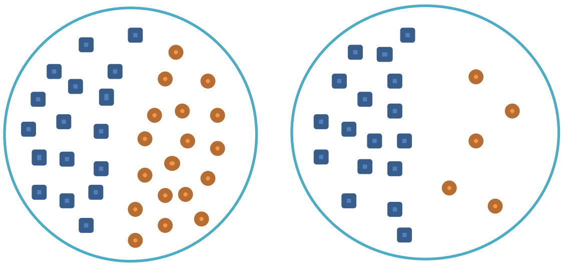

Fig. 1. Balanced versus Imbalanced Data

Extremely unbalanced class distributions [10-12] are a sign of class imbalance issues. A class imbalance may exist in the dataset if samples from one or more classes significantly outnumber those from other classes. Because class imbalance affects the quality and reliability of the output from the machine learning assignment, it has to be addressed using specific techniques and measures. There are three main approaches to addressing the problem of class inequality [10-12]:

1. Data pre-processing level approach

2. Algorithmic level approach

3. Feature selection approach

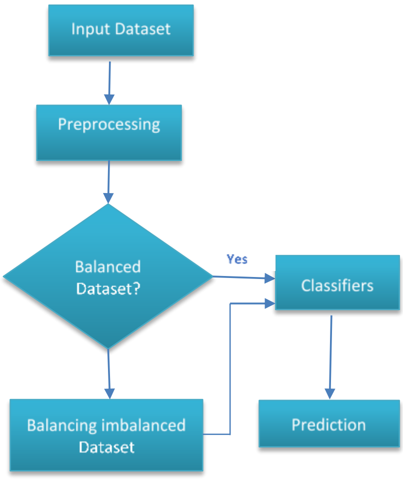

Fig. 2. Framework to handle Class imbalance

4. Reasons for Class Imbalance

It is critical to identify the underlying reasons of this widespread issue before delving into solutions for unbalanced data [13-18]. Comprehending these origins offers significant understanding into the characteristics of unbalanced datasets and guides our strategy for reducing their impacts.

4.1. Data Collection Bias

The main cause of unbalanced data is discrimination in the data gathering process [13]. It is possible for bias to be unintentionally introduced throughout the data gathering process while creating a dataset, be it for image categorization, medical research, or any other sector. This bias can take many different forms, such as undersampling uncommon occurrences and oversampling frequent ones. For example, some diseases may be underestimated in a medical research since they are less common or more difficult to ide.

4.2. Class Imbalance inherent nature of some Datasets

Class inequality [13-18] is a inherent feature of some dataset that has to be solved in particular fields. Think about the situation of detecting spam emails. There is a natural class imbalance because the amount of legitimate emails greatly exceeds that of spam emails. Machine learning models find it difficult to distinguish between the two groups as a result of this imbalance.

4.3. Cost Sensitivity

The underlying cost sensitivity connected to various classes is another element providing unbalanced data. Misclassifying one class can have more negative effects than misclassifying another in several situations. In credit card fraud detection, for instance, the financial institution and the client may suffer large losses in value if a fraudulent transaction is not detected. Misclassifying fraudulent activities therefore comes at a higher cost than misclassifying lawful transactions.

4.4. Uncommon cases incorrectly labeled

Uneven distribution of classes may be introduced by data labeling errors [13-18]. Rare occurrences can occasionally have wrong labels applied, which distorts the distribution of classes. There are a number of reasons why these labeling errors might happen, such as confusion in the definition of class borders or mistakes made by people in the labeling process.

4.5. Favoring one class above other

A phenomena known as "data skewing" occurs when specific sampling strategies inadvertently give preference to one class over another [13-18]. This skewing, which produces a false imbalance in the dataset, can happen during data preparation or an increase.

5. Impact of Imbalanced Data

Unbalanced data poses a significant challenge for machine learning models and their applicability beyond merely being a statistical one. Let's examine a few of the noteworthy effects [13]:

5.1. Models inclined towards majority class

Machine learning models that learned on data that is unbalanced typically display bias in favor of the majority class [13]. During training, the model is exposed to more cases from the majority class, which leads to this bias. As a result, the model could have trouble generalizing to the minority class, which would result in less than ideal performance for that significant subset of the data.

5.2. Failure in detecting infrequent events

In classification problems, sensitivity [13-18] also referred to as the true positive rate—is an essential statistic. It assesses how well the model can recognize good examples. Models frequently show low sensitivity for the minority class in unbalanced datasets, which means they are less successful at identifying and accurately classifying instances from this class. This flaw might have serious repercussions, especially in applications where it's critical to recognize unusual events.

5.3. Misrepresentative Evaluation Metrics

In the case of unbalanced data, classical categorization criteria like accuracy might be inaccurate. Highly accurate algorithms for the majority category may underperform for the minority class, but their total accuracy is still surprisingly high. This mismatch between overall accuracy and performance in a given class might lead to erroneous judgments on the efficacy of the model [13].

5.4. Failure in Generalizing uncommon instances

Unbalanced data might make it more difficult for a model to generalize to new data [13-18]. The model may find it difficult to forecast these situations correctly in the actual world because it was not exposed to the minority class much during training. As a result, the model might not perform well on samples of minority classes that have not been encountered..

5.5. Increased Complexity

In some instances, models trained on imbalanced data may become excessively complex to compensate for the lack of minority class samples. This over-complexity can lead to overfitting, where the model essentially memorizes the training data instead of learning meaningful patterns. Overfit models tend to perform poorly on unseen data, further exacerbating the challenges posed by imbalanced data.

Recognizing these effects is essential to comprehending the seriousness of the unbalanced data problem [13]. It provides significant encouragement for the creation and investigation of methods and approaches to address these problems. The next sections will explore diverse techniques and strategies intended to tackle the intricacies of data that is unbalanced. These techniques seek to bring the dataset back into balance so that predictive models may function more accurately.

6. Different approaches to tackle Class Imbalance Problem

In previous section, we have discussed reasons for uneven distribution of data with its impact on reliability and performance of predictive machine learning algorithms. This section presents methods and strategies to handle this problem. There are three main approaches that may be employed to address the problem of class inequality [13-18]: Resampling, Algorithmic and feature selection

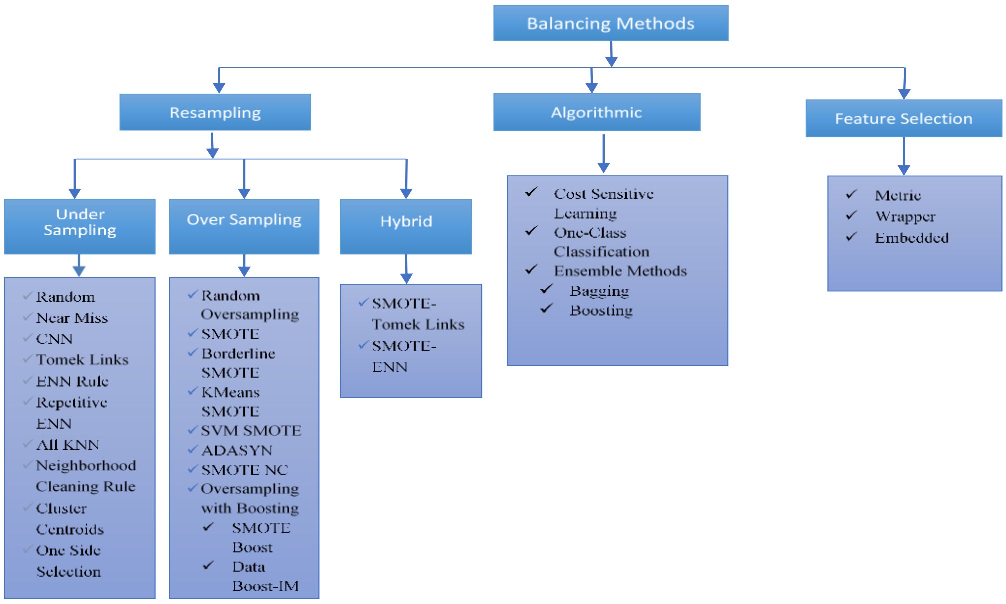

Fig. 3. Types of Balancing Methods

6.1. Techniques of Resampling

To equalize the class distribution, resampling procedures include making modifications to the underlying set of data [18]. Resampling data for analysis is this strategy's primary goal. Prior to an algorithm is trained, a preliminary processing technique called resampling is performed to adjust the class imbalance. Either under-sampling or over-sampling is involved in this procedure.

- Undersampling
- Oversampling
- Combining Undersampling and Oversampling

6.1.1 Undersampling

Decreasing the amount of samples in the category having majority samples is known as under-sampling. A dataset under investigation can be directly under-sampled, and the resulting under-sampled data can be fitted to a machine learning model. Under sampling procedures are typically employed in conjunction with oversampling strategies for the minority class, and this coupling often yields greater effectiveness on the training dataset than employing individual methods separately.

But there are many other algorithms to help us reduce the number of observations in the dataset. These algorithms can be grouped based on their undersampling strategy into[13-19]:

- Prototype generation methods.
- Prototype selection methods.

And within the latter, we find:

- Controlled undersampling
- Cleaning methods

a) Random Sampling

The most basic method of under-sampling is to take random samples from the majority category and eliminate it from the dataset that was used for training [13-20]. It is known as random under-sampling. This method's drawback, despite its simplicity and effectiveness, is that instances are eliminated without consideration for their potential value or significance in establishing the decision border among the classes. This implies that it's likely that important data will be removed.

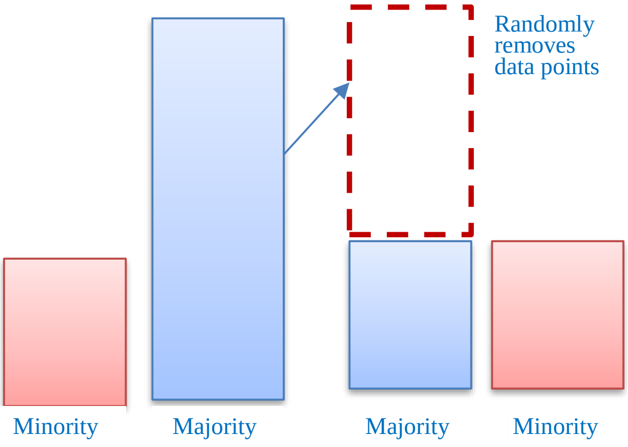

Fig. 4. Random Undersampling

b) Near Miss Undersampling

A well-liked under-sampling technique for unbalanced data is called NearMiss [20-21]. It is a straightforward yet efficient method that keeps the most useful majority class examples while cutting down on the quantity of them. Just a small percentage of the instances from the majority segment that are most similar to the minority category are kept by the algorithm.

NearMiss has several benefits, chief among them being its processing speed and ease of implementation. Additionally, it lowers the possibility of over-fitting and enhances model efficiency. It might not function effectively, though, if the minority class occupies a substantial portion of the spectrum of features.

NearMiss comes in three flavors: NearMiss-1 (Type-1), NearMiss-2 (Type-2), and NearMiss-3 (Type-3). While NearMiss-2 chooses instances that are distant from the majority class, NearMiss-1 chooses instances of the majority category that are nearer to the minority class. Combining the best features of the prior two versions, NearMiss-3 is a hybrid.

c) CNN (Condensed Nearest Neighbors)

In essence, CNN balances the dataset by eliminating occurrences of the majority class without sacrificing any valuable data [20, 21, 22, 26]. CNN's goal is to create a dataset subset that accurately classifies each of the samples in the base dataset. In CNN, first k is selected. It starts by selecting any k points in the store S. It checks whether S is consistent or not. TSC is calculated by using KNN. If consistency is not maintained then it will add samples from dataset such that consistency will be maintained. Procedure is repeated till S becomes consistent.

But this issue can be solved considerably more quickly and effectively with the application of a far more potent technology called Advanced CNN. Condensed Nearest Neighbors is an extremely effective method for balancing an unbalanced dataset with the goal of minimizing information loss from undersampling. However, when instances are eliminated from the dataset, some information is still lost.

d) Tomek Links

Approach was invented by Tomek (1976), and it is one of a variation to the CNN [22]. The Tomek Links technique employs a rule to choose the pair of samples that have certain qualities, as opposed to the CNN approach, which just randomly chooses the data points with its k nearest neighbors from the majority category that wishes to be eliminated:

1. Sample A is nearest neighbor of B
2. Sample B is nearest neighbor of A
3. Examples A and B are from distinct classes—positive and negative, respectively.
4. Remove samples of majority class

This approach may be employed to determine desirable samples of majority category data that is nearest to the minority category data, making it unclear to distinguish and then eliminate the samples from majority class that have the smallest distance calculated by Euclid with the minority category information.

Tomek linkages [22] have the drawback of potentially causing underfitting and a reduction in classification effectiveness if an excessive number of examples from the majority category are removed. To create an evenly distributed dataset, extra approaches like over-sampling the minority category or creating artificial data samples could be required as Tomek links might not be able to exclude every sample from the majority group that is the source of the imbalance. Using Tomek linkages has several benefits: it is easy to use, may effectively reduce noise and improve model accuracy, and can be used to determine significant cases from the minority group.

e) Edited Nearest Neighbor Rule

After identifying the neighbors of the targeted class samples using K-Nearest Neighbors, the edited nearest neighbours approach eliminates samples if any or most of the neighbors are of another category [23, 32].

The following steps are performed by ENN:

- Utilizing the complete dataset, train a K-Nearest Neighbors model.
- Determine the K nearest neighbors of every sample (just for the selected classes).
- If most or all of an observation's neighbors are in a different class, remove the observation.

f) Repetitive Edited Nearest Neighbors

By repeating the procedure several times, repeated ENN expands ENN [32, 33]. Additional data will often be deleted if the algorithm is run again. The parameter selection allows the user to choose how many times the modified closest neighbor's algorithm should be run .

The repetitions will come to an end when:

- It appears that the number of repeats has reached its maximum, or
- no further sample are eliminated, or
- A minority category emerges from one of the majority groups, or
- A majority class vanishes during the undersampling process.

g) All KNN

In the All-KNN variant of the Repeated ENN [33, 42], more neighbors are assessed during each ENN round [32]. Each time iterating, the neighborhood is increased by 1 starting with editing based on 1-Nearest Neighbor**.** When the user-specified largest amount of neighbors to analyze reaches to its limit, or when the majority class shifts to the minority class, AllKNN ceases to clean.

h) Neighborhood Cleaning Rule

When undersampling the imbalanced datasets, the Neighborhood Cleaning Rule, or NCR, addresses the positive (majority) and negative (minority) samples independently. For each occurrence in the data set, NCR first determines its three closest neighbors. Next, it takes the following actions [32, 33]:

- The selected sample is eliminated if it is a member of the majority class and the categorization provided by its three closest neighbors differs from the class of the selected sample.
- The closest neighbors who are members of the majority class are eliminated if the sample is in the minority class and its 3 closest counterparts have incorrectly categorized it.

i) Cluster Centroids

K-means is used by Cluster-Centroids to decrease the sample size. Consequently, the centroids of the K-means algorithm will be used to represent each class rather than the actual samples. This approach creates an undersampled set by using clustering techniques to create a new set on the basis of centroids. The KMeans process's cluster centroid is used by the procedure to create a new set. This technique uses KMeans strategy to undersample the majority class from a cluster of majority samples [33].

Upon the identification of the majority class's cluster centroid, it makes the subsequent decisions:

- Statistically insignificant instance is the one that belongs to the majority group and is located furthest from the cluster centroid.
- The most significant instance is the one that belongs to the majority category and is located closest to the cluster centroid.

j) One Sided Selection

This approach merges the CNN rule with Tomek Links [28, 33]. In contrast to CNN, which eliminates distant instances from the majority class, Tomek connections eliminate the noisy and ambiguous cases.

k) Instance Hardness Threshold

By eliminating hard samples, the Instance Hardness Threshold (IHT) undersampling technique helps to reduce class imbalance [34]. Samples that have little chance of being categorized will be eliminated from the dataset. The model will then be constructed using the data that was under sampled

Table 1. Comparison of Undersampling Methods

Method	Methodology	Advantages	Disadvantages
Random Undersampling	- Randomly removes data points from majority class	- Reduces data samples - Simple and Fast	- Threat of excluding useful samples
Near Miss Undersampling	- Focuses on keeping samples that are useful, excluding redundant and noisy. - Capability to properly identify minority class	- Reduces majority samples - Computationally efficient - Preserves information of majority category - Ease of Implementation	- Reduces overfitting
CNN (Condensed Nearest Neighbors)	- Focuses on reducing information loss - Uses Basic approach of NN	- Initially samples are selected randomly - Retention of unnecessary instances	- Does not promise smallest subclass - Information loss
Tomek Links	- Removes majority samples from class boundaries. - Identifies significant samples from minority class	- Increases classification accuracy - Simple and effective in reducing noise	- Removing too many samples results in underfitting
Edited Nearest Neighbor Rule	- Identifies and removes redundant and noisy elements from majority class	- Noise Reduction - Improved Generalization - Safeguarding of Decision Boundaries	- Some information is lost
Repetitive Edited Nearest Neighbors	- Repeats ENN many times	- Supports multi-class sampling - Repeating algorithm removes more data.	- Loss of Significant information
Neighborhood Cleaning Rule	- Incorporates CNN and ENN	- Focuses more on data cleaning than reduction	- Loss of Information
Cluster Centroids	- Forms clusters of majority class and replaces with centroids	- reduces the size of the majority category while maintaining an initial distributions	- Original Information is lost
One Sided Selection	- Combines Tomek Links and CNN Rule.	- Reduces execution time and storage problems - Supports multi-class sampling	- OSS does not consider that subsets exit in class,
Instance Hardness Threshold	- Identifies hard instances (Lower probability samples)	- Retains samples with higher class probabilities.	- Only highest probability samples are retained

6.1.2 Oversampling

a) Random Oversampling

The most straightforward oversampling method for balancing the dataset's inequality is random oversampling [33]. By duplicating the minority class instances, it evens out the data. The dataset is susceptible to overfitting as a result of the duplicated data, yet no information is lost.

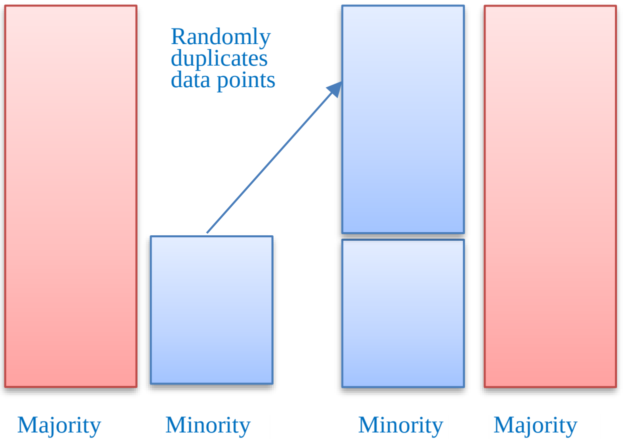

Fig. 5: Random Oversampling

b) SMOTE

Some of the most often used over-sampling method, SMOTE was created by Chawla et al. [18, 19, 33]. The derived instances are distinct from the initial minority class because SMOTE creates instances according to the distance of each data, usually employing the Euclid distance and the minority class nearest neighbors, as opposed to random oversampling, which simply replicates several random instances from the minority group.

The steps involved in creating the synthetic instances are as follows, to put it briefly [18, 19].

- Select data at random coming from the minority class.
- Determine the random data's Euclidean distance to its k closest neighbors.
- Add the outcome to the minority class as a fabricated sample after multiplying the difference by a random value between 0 and 1.
- Continue the process until the targeted minority class percentage is reached.

Unlike the preliminary oversampling approach, this approach adds new "information" to the data since the artificially constructed data are reasonably near to the feature set of the minority category.

Benefits of SMOTE

In order to increase model performance, SMOTE may create new samples based on pre-existing ones, which helps to expand the dataset's amount of information [19].

SMOTE's Limitations

The potential for noise introduction with artificial instances is one of SMOTE's primary disadvantages, particularly when the number of nearest neighbours is set very high [33]. Furthermore, examples of the minority class that are sparsely distributed or closely clustered may not respond well to SMOTE.

c) Borderline Smote

The bridges of minority class instances are generated because the region of majority class points contains some minority points or outliers. This is an issue with Smote, and Borderline Smote is used to resolve it. Based on SMOTE, borderline SMOTE is an enhanced oversampling approach that improves the sample category distribution by combining fresh samples using only a small number of class samples on the boundary [31, 33, 64].

Simply the minority cases close to the boundary are oversampled in the Borderline Smote approach. The minority class points are classified as border points and noise points by it. Border points have both majority and minority class points in their neighbor, whereas noise points are minority class points with the majority of points in their neighbor. By utilizing only these boundary points and ignoring the noise points, the Borderline Smote algorithm seeks to generate synthetic points.

d) KMeans SMOTE

A technique for oversampling class-imbalanced data is K-MeansSMOTE. By producing minority class samples in critical and secure regions of the input space, it facilitates categorization. The technique successfully addresses imbalances between and within classes while avoiding noise production.

There are 5 stages in the K-Means SMOTE process [33, 35]:

- Apply the k-means clustering method to cluster all of the data.
- Choose clusters with a large proportion of samples from minority classes.
- Where samples from minority classes are dispersed widely, add extra artificial samples to the clusters.

SMOTE is used to oversample each reduced cluster.

e) SVM SMOTE

A substitute to Borderline SMOTE is SVM-SMOTE, in which the support vectors of a support vector machine that separates the classes are utilized as the data points to generate the synthetic data [35, 36, 64]. By interpolation between each minority class support vector and its closest neighbors, artificial data is produced. As a result, at the decision border, additional data points are produced.

f) Adaptive Synthetic Sampling (ADASYN)

In order to solve the problem of producing synthetic samples in areas of the feature space that are less distant from the selection boundary, ADASYN [27, 33] is an alternate oversampling method. It functions by creating more artificial examples for minority class samples, or individuals that are more near the decision border, which are harder to learn.

If some of their nearest neighbors belong to a different class, ADASYN utilizes those instances as templates for the artificial data samples from the minority class. It is more probable to be utilized as a template if it has more neighbors who belong to the opposing class. It first chooses the templates and then uses interpolation to create examples based on the template's closest neighbors inside the same class.

g) SMOTE NC

The synchronous oversampling approach is limited to datasets that contain all continuous features. Smote-NC (Nominal and Continuous) is a variant of Smote for a dataset containing categorical characteristics [36, 67].

By one-hot encoding, Smote may also be utilized for data including categorical characteristics, albeit this could lead to a spike in dimensionality. Although label encoding can be additionally employed to transform categories to numbers, it may produce extraneous information in the process. For this reason, in situations where there is mixed data, SMOTE-NC must be used. By indicating the categorical characteristics, Smote-NC may be employed. As opposed to producing synthetic data, Smote would resample the data that is categorical.

h) Oversampling with Boosting

SMOTEBoost

Chawla et. al. proposed this method. With this approach, SMOTE is incorporated into every boosting cycle. That is, the SMOTE is used to produce some fresh synthetic minority cases prior to the creation of any more weak learners. This has a number of benefits [42]:

- More attention is paid to the minority class by each subsequent classifier.
- More variety is produced since every classifier is based on a distinct data sample.

DataBoost-IM

The DataBoost-IM [43] was developed by Hongyu Guo and Viktor as an extension of their previous DataBoost, which was designed to handle balanced datasets, for unbalanced data. With this method, the minority class receives greater attention from the classifiers, in addition to concentrating on cases that were incorrectly categorized.

Table 2. Comparison of Undersampling methods

Method	Methodology	Advantages	Disadvantages
Random Oversampling	- Naïve technique - Replicates randomly selected samples from minority group	- Widely used - Simple and Fast	- Overfitting - Increase in training time
SMOTE	- Synthetically generates samples of minority class using KNN approach - Generates samples uniformly	- Works well with low dimensional data - Improves classification performance	- Overfitting and Noisy Samples - Over generation minority class space - Less attractive for high dimensional data
Borderline SMOTE	- Replicates minority samples near the borderline	- Performs better than SMOTE	- Does not identify samples near decision boundary
K-Means SMOTE	- Addresses both between class imbalance and within class imbalance - Groups samples by K-Means and generation using SMOTE	- Avoid generation of noisy data	- Estimation of parameter k
SVM SMOTE	- Uses Support vectors to find new samples - focuses on increasing minority points along the decision boundary.	- Increases the chance to see minority samples near the boundary - Overcomes limitations of SMOTE	- Works well when low degree of overlap between classes
ADASYN	- Emphases on minority samples that are hard to classify - Assigns weights to minority samples	- No fixed oversampling ratio	- Fails to identify noisy instances - Susceptible to outliers

			- Fails to discover all minority samples on decision boundary.
SMOTE NC	- Works well for categorical data	- Suitable for mixed data	- Not performing well if all the samples are of nominal type

6.1.2 Combination of Undersampling and Oversampling

Oversampling and Undersampling approaches can be combined to create to increase performance. SMOTE combined with Tomek Links undersampling and SMOTE combined with ENN are two examples.

SMOTE-Tomek Links

This method involves first oversampling the minority class using the SMOTE method to get an evenly distributed data, and then identifying and removing samples from the majority classes using Tomek Links. For a binary classification job, the combination was demonstrated to reduce false negatives at the expense of increasing false positives [66].

SMOTE-ENN

The ENN methodology is a data filtering method that eliminates unimportant and noisy instances from the dataset. SMOTENN is able to produce synthetic samples that are less noisy and more representative of the minority class by combining SMOTE and ENN. This may result in the model performing better overall.

6.2 Algorithmic Approaches

a) Cost Sensitive Learning

The penalty of incorrectly identifying infrequent events is frequently far larger in real-world situations [54, 55]. When it comes to fraud detection, mistaking an act of fraud for a genuine one can lead to monetary losses and reputational harm for a business. By taking into account the varied penalties related to different kinds of incorrect classifications, cost-sensitive learning enables us to solve the issue of inequalities in classes and improve the effectiveness of models. Within the field of algorithms for learning, cost-sensitive learning recognizes that misclassification mistakes in unbalanced datasets come with different consequences. It emphasizes changing the tuning functions of models to reduce the total cost of wrongly classifying data rather than the total error rate. Misclassification costs in CSL fall into four separate groups: true positive, true negative, false positive, and false positive. The expenses connected with each accurate and inaccurate categorization are shown in a cost matrix. As a result, the cost matrix records these various expenses and directs the learning process appropriately. Cost needs to be considered in minimization function of model.

b) One Class Classification

As a data preparation technique, identifying and eliminating outliers may be necessary for fitting a machine learning model. One of the methods for detecting outliers is One-Class Classification (OCC) [56]. OCC involves training a model on normal data and predicting whether candidate sample is normal or outlier. OCC is trained on data that have samples only form normal class. OCC is used for two-mode classification problems where positive samples are considered as normal and negatives samples are regarded as outlier.

6.3 Ensemble Methods

The algorithms of one model operate well in classification, but they suffer from bias due to a predetermined set of attributes. Ensemble learning can be used to reduce this kind of bias. . Several classifiers collaborate to learn the dataset that was used in ensemble class.

Table 3. Bagging versus Boosting

	Bagging	Boosting
Methodology	Aggregates many models that have been trained on various data subsets	Train models one after the other, paying attention to the mistakes produced by the earlier model.
Aim	By averaging each individual model mistake, to lower variation.	Corrects the preceding model's misclassifications, reducing bias as well as variance.
Sampling	To generate data subsets, use Bootstrapping.	Reweights the data according to the prior model's mistake, causing the subsequent models to concentrate on cases that were incorrectly.
Model Weight	Every model has the same weight when it comes to the ultimate choice.	Accuracy is the basis for model weighting.
Dealing With Errors	The error rate is the same for all models.	Assigns more weights to samples having greater error rate.

Overfitting	Fewer overfitting cases because of the average process.	resistant to overfitting
Accuracy	Decreases variation, which increases accuracy.	Decreases both bias as well as deviation to produce greater accuracy.
Types of Methods	Random Forest (RF)	AdaBoost, Gradient Boost, XGBoost

a) Bagging

Bagging is a combination method that is frequently used in classification instances. Its superiority over random sampling stems from its ability to combine individual classifications and increase classification accuracy [51, 52, 53]. Here are a few bagging strategies mixed with resampling approaches

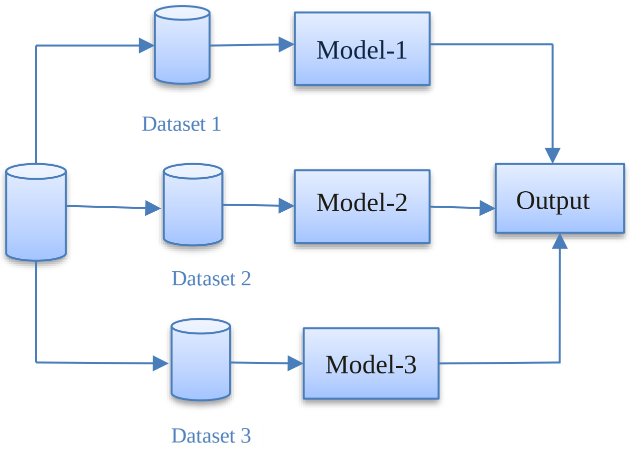

Fig. 6 Bagging

UnderBagging

To equalize the distribution of classes, UnderBagging employs random under-sampling to decrease the number of majority samples in every bag of Bagging [51].

OverBagging

UnderBagging uses a bootstrapping mechanism to decrease the majority class, whereas OverBagging uses one to increase or add minority classes [51].

SMOTE-Bagging

Chawla [51] proposed the use of SMOTE in combination with bagging as a technique for oversampling.

UnderOverBagging

Although the data production procedure differs from that of the UnderBagging or OverBagging algorithms [51], UnderOverBagging is a mix of approaches from undersampling, oversampling, and bagging. UnderOverBagging and SMOTEBagging have a similar data creation technique.

BEV

BEV is a part of the UnderBagging group, which combines bootstrap aggregating with sample undersampling [51, 57]. The goal of BEV is to get over the challenges that occur with categorization, particularly in classes when there aren't enough variations in the bag.

RBB

One of method for dealing with uneven classes is roughly balanced bagging. Furthermore, Hido and Kashima's original UnderBagging approach includes Roughly Balanced Bagging [51]. This method works well for bringing each class's average back into balance.

Table 4. Comparison of Bagging Algorithms

Methods	Demerits	Merits
UnderBagging	- Uses undersampled versions of data - Low performance with high majority to minority class ratio. - Sensitive to highly unbalanced classes	- Shows better performance compared to traditional classification trees
OverBagging	- Uses oversampling - Sensitive to Outliers	- Increased prediction correctness of minority class
SMOTE-Bagging	- Weak with high majority to minority class ratio - Sensitive to Outliers	- Increases prediction precision of minority
UnderOverBagging	- Sensitive to Outliers	- Outcome is better for small number of trees
Bagging Ensembles Variation	- Weak with high majority to minority class ratio	- Improves prediction accuracy
Roughly Balanced Bagging	- Low performance with large number of trees.	- Improves prediction accuracy of minority - Better on different classifiers and diverse

		data - Resistant to Outliers

b) Boosting

Predictive models both efficiency and precision can be enhanced by the use of sophisticated ensemble learning methods called boosting algorithms. All boosting algorithms operate by using errors produced by the weak learning method that was used to train the preceding model to learn from them and try to avoid making the same errors. Combining several weak learners into one powerful learner is known as "boosting."

AdaBoost (Adaptive Boosting)

Decision trees are the basic learners used by AdaBoost, and every iteration's weights are updated in accordance with the exponential loss function [44]. AdaBoost's primary concept is to concentrate more on samples that prior learners incorrectly categorized and less on examples that were successfully classified.

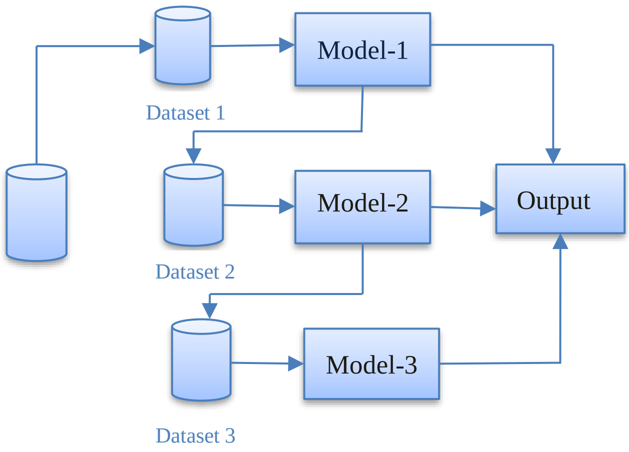

Fig. 7. Boosting Process

Table 5. Comparison of Boosting Methods

Method	Advantages	Disadvantages
AdaBoost	- Simple, interpretable - Basics of Boosting - Good for smaller and medium size datasets	- Needs high-quality dataset - Sensitive to outliers and noise - No parallelism
Gradient Boosting	- Accuracy and Performance - Can be used in many applications - Well implemented	- Requires pre-processing like one hot encoding - More training time - Overfitting - Hard to interpret
XGBoost	- Better Regularization and stability - Scalable and efficient - Handles Large Scale data - Prevents overfitting	- Moderate memory use - Designed for tabular data - Computationally complex greedy algorithm
LightGBM	- Handles Categorical data - Regularization - Low memory usage, Better accuracy	- Constructs big tree structure - Interpretation difficulty
CatBoost	- Works well on Categorical data - More accurate on categorical data - Regularization	- Memory usage high - High Training time

Gradient Boosting

In this model, errors of previous models are recognized by gradient. To improve accuracy, the capacity for prediction of each classifier is kept constant by limiting its learning rate [47, 50].

XGBoost

It is a more aggressive variation of the earlier gradient boosting technique. The primary distinction between XGBoost and Gradient-Boosting is that the latter makes use of a regularization approach [49, 50]. Put simply, it is an algorithm that regularizes the already-existing gradient-boosting form.

LightGBM (Light Gradient Boosting Machine)

Decision trees in the LightGBM algorithm [48, 50] are developed leaf wise, which means that just one leaf out of the entire tree will develop at a time. The GOSS approach is used in LightGBM to sample the data for the purpose of training the decision tree. Using this procedure, all data samples' variances are computed and arranged in descending order. Low variance samples given less weightage.

CatBoost

CatBoost's growing decision tree makes it superior than others. It uses symmetric decision trees in its growth process [46, 50].

7. Performance Evaluation Metrics for Class Imbalanced Datasets

The performance of class imbalance control techniques may be assessed and compared using a variety of measures, including feature importance, cross-validation, accuracy, precision, recall, F1-score, AUC-ROC curve, AUC-PR curve, MCC Score, and confusion matrix [13]. Their applicability may be tested and compared across different datasets and circumstances. In these situations, traditional measurements like accuracy might be deceptive since they don't give a whole picture of a model's efficacy.

Recall and precision are crucial in datasets with unequal distributions because strong recall guarantees that critical samples are not overlooked while precision concentrates on making accurate positive predictions. A compromise between these two measurements is offered by the F1-score. The metrics AUC-ROC and AUC-PR are useful for evaluating binary classifiers at different threshold levels. MCC is a reliable statistic that offers a comprehensive evaluation of categorization performance. In actuality, it is best to take into account a variety of indicators and the trade-offs between them in order to have a thorough grasp of a model's performance on unbalanced data.

8. Conclusion

A prevalent and intricate problem in machine learning is unbalanced data, which can have significant effects on model performance and practical applications. This work has offered an in-depth examination of the complex terrain of unbalanced data, including everything from its underlying origins and effects to a detailed investigation of approaches and assessment metrics intended to address this issue. To help academics and practitioners deal with unbalanced data in real-world machine learning projects, useful guidelines have been provided. In order to customize solutions to particular datasets and issue situations, these guidelines stress the need of experimentation, parameter adjustment, and the investigation of various methodologies.

References

[1] Iqbal H. Sarrkar, “Machine Learning: Algorithms, Real-World Applications and Research Directions.” SN Computer Science, 2:160, 2021.

[2] Mahdavinejad MS, Rezvan M, Barekatain M, Adibi P, Barnaghi P, Sheth AP, “Machine learning for internet of things data analysis: a survey”, Digit Commun Netw. Vol. 4, issue 3:161–175, 2018.

[3] K. Shailaja, B, Seetharamulu , M. A. Jabbar, "Machine Learning in Healthcare: A Review," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, pp. 910-914, 2018.

[4] Hafsa Habehh, Sunil Gohel, “Machine in Healthcare”, Current Genomics, vol. 22, issue 4, pp, 291-300, Dec. 2021.

[5] Dara, S., Dhamercherla, S., Jadav, S.S. et al., “ Machine Learning in Drug Discovery: A Review”, Artif Intell Rev 55, 1947–1999 2022.

[6] Daniel Broby,"The use of predictive analytics in finance" ,The Journal of Finance and Data Science,Volume 8, pp. 145-161, 2022,

[7] Omer Artun, Domnique Levin, “Predictive Marketing: Easy Ways Every Marketer Can Use Customer Analytics and Big Data”, Wiley Publications, 2015.

[8] Seyedan, M., Mafakheri, F,.” Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities.", Journal of Big Data **7**, 53, 2020.

[9] Marzieh Fathi, Mostafa Haghi Kashan, Seyed Mahdi Jameii, · Ebrahim Mahdipour, “ Big Data Analytics in Weather Forecasting: A Systematic Review ”, Archives of Computational Methods in Engineering, Jun 2021.

[10] Sun, Y., Wong, A. K. C., and Kamel, M. S., “Classification of imbalanced data: a review,” International Journal of Pattern Recognition and Artificial Intelligence, vol. 22, no. 4, pp. 687–719, 2009.

[11] Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, Gong Bing, "Learning from class-imbalanced data: Review of methods and applications" , Expert Systems with Applications,Volume 73, pp. 220-239, 2017.

[12] Salim Rezvani, Xizhao Wang, "A broad review on class imbalance learning techniques" , Applied Soft Computing, Volume 143, 2023,

[13] Mukhtar shah, “Imbalanced Data in Machine Learning: A Comprehensive Review”, Department of Machine Learning, University of Jumeirah.

[14] Barandela, R., Sánchez, J. S., García, V., & Rangel, E., “Strategies for Learning in Class Imbalanced Datasets. Pattern Recognition”, 36(3), 849-851, 2003

[15] Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., “Learning from Class-imbalanced Data:

Review of Methods and Applications", Expert Systems with Applications", 73, 220-239, 2017

[16] Japkowicz, N., Stephen, S., "The Class Imbalance Problem: A Systematic Study", Intelligent Data Analysis, 6(5), 429-449., 2002

[17] Kubat, M., Matwin, S., "The Class Imbalance Problem: A Systematic Study", Intelligent Data Analysis, 2(3), 429-449, 1998.

[18] Chawla, N., et al., "Special issues on learning from imbalanced data sets," ACM SigKDD Explorations Newsletter, vol. 6, no. 1, pp. 1–6, 2004

[19] Chawla, Nitesh V., et al., "SMOTE: Synthetic Minority Over-Sampling Technique," Journal of Artificial Intelligence Research, vol. 16, no. 1, pp. 321–357, 2002.

[20] Wongvorachan T, He S, Bulut O., "A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining.", *Information*, 14(1):54., 2023

[21] Akira Tanimoto, So Yamada, Takashi Takenouchi, Masashi Sugiyama, Hisashi Kashima, "Improving imbalanced classification using near-miss instances", Expert Systems with Applications,Volume 201,2022,.

[22] Tomek, I., "Two Modifications of CNN", IEEE Transactions on Systems, Man, and Cybernetics (SMC-6): 769-772, 1976

[23] Batista, G. E., Prati, R. C., and Monard, M. C., "A study of the behavior of several methods for balancing machine learning training data.", ACM SIGKDD Explorations Newsletter, 6(1):20–29, 2004

[24] Nitesh V Chawla, Aleksandar Lazarevic, Lawrence O Hall, and Kevin W Bowyer, "Smoteboost: Improving prediction of the minority class in boosting", In European Conference on Principles of Data Mining and Knowledge Discovery, pages 107–119. Springer, 2003.

[25] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao, "Borderline-smote: a new over-sampling method in imbalanced data sets learning" In International Conference on Intelligent Computing, pp. 878–887, Springer, 2005.

[26] P. Hart, "The condensed nearest neighbor rule", IEEE Trans. Inf.bTheor., 14(3):515–516, September 2006..

[27] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning", In 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), pp. 1322–1328, 2008.

[28] Miroslav Kubat, Stan Matwin, et al. "Addressing the curse of imbalanced training sets: one-sided selection", In ICML, volume 97, pages 179–186. Nashville, USA, 1997.

[29] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou, "Exploratory undersampling for class-imbalance learning", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2):539–550, 2009.

[30] Inderjeet Mani and I Zhang, "knn approach to unbalanced data distributions: a case study involving information extraction", In Proceedings of workshop on learning from imbalanced datasets, 2003.

[31] Hien M Nguyen, Eric W Cooper, and Katsuari Kamei, "Borderline over-sampling for imbalanced data classification", International Journal of Knowledge Engineering and Soft Data Paradigms, 3(1):4–21, 2011

[32] Dennis L Wilson, " Asymptotic properties of nearest neighbor rules using edited data", IEEE Transactions on Systems, Man, and Cybernetics, (3):408–421, 1972.

[33] Ajiknya More, " Survey of resampling techniques for improving classification performance in unbalanced datasets", Computer Science Mathematics, arXiv.org, 2016

[34] Smith, M.R., Martinez, T., Giraud-Carrier, C. An instance level analysis of data complexity", Mach Learn 95, 225–256, 2014

[35] Bagui, S.S., Mink, D.; Bagui, S.C., Subramaniam, S, "Determining Resampling Ratios Using BSMOTE and SVM-SMOTE for Identifying Rare Attacks in Imbalanced Cybersecurity", Data. Computers 2023, 12, 204.

[36] D. Utari, "Integration of SVM AND SMOTE-NC for classification of Heart Failure", Barkekeng: J. Math. & App., vol. 17, no. 4, pp. 2263-2272, Dec. 2023.

[37] V. S. Gaikwad, S. S. Deore, G. M. Poddar., R. V. Patil,, D. S. Hirolikar, M. P. Borawak.S. K. Swarnkar,"Unveiling Market Dynamics through Machine Learning: Strategic Insights and Analysis.", International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 388–397, 2024

[38] Tarambale, M., Naik, K., Patil, R. M., Patil, R. V., Deore, S. S., & Bhowmik, M. "Detecting Fraudulent Patterns: Real-Time Identification using Machine Learning", International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 650–.660, 2024

[39] Wei-Chao Lin, Chih-Fong Tsai, Ya-Han Hu, Jing-Shang Jhang, "Clustering-based undersampling in class-imbalanced data",Information Sciences,Volumes 409–410, pp. 17-26, 2017

[40] P. S. Patil, S. R. Kolhe, R. V. Patil, P. M. Patil ,"The Comparison of Iris Recongition using Principal Component Analysis, Log Gabor and Gabor Wavelets", International Journal Of Computer Applications, Vol-43, No. 1., pp. 29-33, 2012

[41] R. V. Patil and K. C. Jondhale, "Edge based technique to estimate number of clusters in k-means color image segmentation", 2010 3rd International Conference on Computer Science and Information Technology, Chengdu, China, pp. 117-121, 2010

[42] Chawla, N. V., Lazarevic, A., Hall, L.O., Bowyer, K.W, " SMOTEBoost: Improving Prediction of the Minority Class in Boosting", Lecture Notes in Computer Science, vol 2838. Springer, Berlin, Heidelberg, pp. 107-109, 2003

[43] Hongyu Guo, Herna L Viktor, "Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach", Sigkdd Explorations., vol. 6, issue 1, pp. 30-39, 2004

[44] Freund, Y. and Schapire, R.E., "A decisiontheoretic generalization of on- line learning and an application to boosting", Journal of Computer and System Sciences, Vol. 55, Issue 1, Pages 119-139, 1997.

[45] Chengsheng, T., Huacheng, L., Xu, B., "AdaBoost typical Algorithm and its application research", MATEC Web of Conferences, Vol. 139, Issue 2, 00222, France, 2017

[46] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V., Gulin, A., "CatBoost: unbiased boosting with categorical features", NeurIPS - 32nd Conference on Neural Information Processing Systems, Montreal, pp, 6638-6648, 2018.

[47] Friedman, J.H. "Stochastic gradient boosting", Computational Statistics & Data Analysis, Vol. 38, Issue 4, pp. 367-378, 2002.

[48] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y., "LightGBM: a highly efficient gradient boosting decision tree", NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc. California, pp. 1-9, 2017.

[49] Ma, J., Zhongqi, Y., Qu, Y., Xu, J., Cao, Y., "Application of the XGBoost Machine Learning Method in PM2.5 Prediction: A Case Study of Shanghai", Aerosol and Air Quality Research, Vol. 20, Issue 1, Pages 128-138, 2019.

[50] Korau Soskun, Gürcan Çetin, "A comparative evaluation of the Boosting Algorithms for Network Classification", International Journal of 3D Printing and Digital Technologies, 6(1), 101-112, 2022.

[51] B Lukmanul Hakim; Bagus Sartono; Asep Saefuddi, "Bagging Based Ensemble Classification Method on Imbalance Datasets", International Journal of Computer Science and Network, pp. 670-676, 2017

[52] R. Barandela, R. M. Valdovinos, and J. S. S´anchez, "New applications of ensembles of classifiers," Pattern Anal. App, Vol. 6, pp. 245–256, 2003.

[53] J. Blaszczynski , J. Stefanowski, Szajek, "Local Neighbourhood in Generalizing Bagging for Imbalanced Data", COPEM ECML-PKKD. Workshop Proceedings. Solving Complex Machine Learning Problems with Ensemble Methods, 2013

[54] N. Thai-Nghe, Z. Gantner and L. Schmidt-Thieme, "Cost-sensitive learning methods for imbalanced data," *The 2010 International Joint Conference on Neural Networks (IJCNN),* Barcelona, Spain, pp. 1-8,2010

[55] Ibomoiye Domor Mienye, Yanxia Sun, "Performance analysis of cost-sensitive learning methods with application to imbalanced medical data", Informatics in Medicine Unlocked, Volume 25, 2021,

[56] Hayashi, T., Fujita, H, "One-class ensemble classifier for data imbalance problems", Appl Intell 52, 17073–17089, 2022.

[57] C. Li, "Classifying Imbalanced Data Using A Bagging Ensemble Variation (BEV)", Conference: Proceedings of the 45th Annual Southeast Regional Conference, pp. 203-208, March 2007.

[58] Ramyachitra D. Manikanda P, " Imbalanced Dataset Classification And Solutions: A Review" International Journal of Computing and

[59] N. Thai-Nghe, Z. Gantner and L. Schmidt-Thieme, "Cost-sensitive learning methods for imbalanced data," The 2010 International Joint Conference on Neural Networks (IJCNN), Barcelona, Spain, pp. 1-8,2010

[60] Shaza M Abd Elrahman1 and Ajith Abraham, "A Review of Class Imbalance Problem" Journal of Network and Innovative Computing. Vol. 1, pp. 332-340, 2013.

[61] Rajendra V. Patil, R. Aggarwal, "Comprehensive Review on Image Segmentation Applications", Sci.Int.(Lahore), 35(5), pp. 573-579, Sep. 2023

[62] Patil, R. V., & Aggarwal, R., "Edge Information based Seed Placement Guidance to Single Seeded Region

Growing Algorithm.”, International Journal of Intelligent Systems and Applications in Engineering, 12(12s), 753–759, 2024

[63] Patil, R. V. ., Aggarwal, R. ., Poddar, G. M. ., Bhowmik, M. ., & K. Patil, M. , “Embedded Integration Strategy to Image Segmentation Using Canny Edge and K-Means Algorithm”, International Journal of Intelligent Systems and Applications in Engineering, 12(13s), 01–08. 2024

[64] Nemade, B. ., Bharadi, V. ., Alegavi, S. S., & Marakarkandy, B., “ A Comprehensive Review: SMOTE-Based Oversampling Methods for Imbalanced Classification Techniques, Evaluation, and Result Comparisons”, International Journal of Intelligent Systems and Applications in Engineering, 11(9s), 790–803, 2023

[65] Hui Han, Wen-Yuan Wang & Bing-Huan Mao, "Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning,” International Conference on Intelligence Computing and Intelligent Systems (ICIS), 2005.

[66] L. Demidova and I. Klyueva, "SVM classification: Optimization with the SMOTE algorithm for the class imbalance problem," 2017 6th Mediterranean Conference on Embedded Computing (MECO), Bar, Montenegro, pp. 1-4, 2017.

[67] C. Bunkhumpornpat, c. Lursinsap, "Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-Sampling Technique for Handling the Class Imbalanced Problem", Lecture Notes in Computer Science, vol 5476. Springer, Berlin, Heidelberg.

[68] M. Mukherjee and M. Khushi, “SMOTE-ENC: A Novel SMOTE-Based Method to Generate Synthetic Data for Nominal and Continuous Features,” Applied System Innovation, vol. 4, no. 1, p. 18, Mar. 2021

[69] Triguero, S. García, M. Galar, J. A. Sáez, and F. Herrera, "Enhancing techniques for learning decision trees from imbalanced data," Knowledge-Based Systems, vol. 87, pp. 69-81, 2015.

[70] Mikel Galar, Fransico, “A review on Ensembles for the class Imbalance Problem: Bagging, Boosting and Hybrid Based Approaches” IEEE Transactions On Systems, Man, And Cybernetics—Part C: Application And Reviews, Vol.42, No.4 July 2012.

[71] Gaikwad, V. S., Shivaji Deore, S., Poddar, G. M., R. V. Patil,, Sandeep Hirolikar, D. ., Pravin Borawake, M. ., & Swarnkar, S. K. . Unveiling Market Dynamics through Machine Learning: Strategic Insights and Analysis. International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 388–397. 2024

[72] Tarambale, M. , Naik, K, Patil, R. M. , Patil, R. V. , Deore, S. S. , & Bhowmik, M. Detecting Fraudulent Patterns: Real-Time Identification using Machine Learning. International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 650 –.660, 2024