Çukurova Üniversitesi Mühendislik Fakültesi Dergisi, 39(2), ss. 489-501, Haziran 2024
Cukurova University Journal of the Faculty of Engineering, 39(2), pp. 489-501, June 2024

A Comparative Study on Data Balancing Methods for Alzheimer's Disease Classification

Esma ÖTER[1] ORCID *0009-0007-9823-2836*
Yahya DOĞAN*[1] ORCID *0000-0003-1529-6118*

[1]Siirt University, Faculty of Engineering, Department of Computer Engineering, Siirt, Türkiye

Geliş tarihi: 07.01.2024 Kabul tarihi: 27.06.2024

Atıf şekli/ How to cite: ÖTER, E., DOĞAN, Y., (2024). A Comparative Study on Data Balancing Methods for Alzheimer's Disease Classification. Cukurova University, Journal of the Faculty of Engineering, 39(2), 489-501.

Abstract

Alzheimer's disease is a prevalent neurological disorder affecting millions of people worldwide, often associated with the aging process, leading to the death of nerve cells in the brain and loss of connections. Recently, promising results have been demonstrated in diagnosing Alzheimer's disease using deep learning models, and various approaches for early diagnosis have been proposed. However, the imbalance in health datasets, particularly those containing rare cases, can lead to performance losses and misleading results during model training. This study focuses on these imbalance issues, evaluating the effectiveness of different balancing methods using the Alzheimer's MRI dataset. In this context, the performance of SMOTE, ADASYN, and Weight Balancing methods is compared using a custom model. Experimental results indicate that, compared to the original imbalanced dataset, Weight balancing outperforms in terms of accuracy, precision, recall, and F1 score. While SMOTE and ADASYN show improvement in various metrics, they are considered inferior to the Weight Balancing method. This study contributes to selecting data-balancing methods to enhance the accuracy of deep learning models in Alzheimer's disease classification and emphasizes the importance of addressing class imbalances in health datasets.

Keywords: Deep learning, Convolutional neural networks, SMOTE, ADASYN, Weight balancing

Alzheimer Hastalığı Sınıflandırması için Veri Dengeleme Yöntemlerinin Karşılaştırmalı Bir Çalışması

Öz

Alzheimer hastalığı, dünya genelinde milyonlarca insanı etkileyen yaygın bir nörolojik bozukluktur ve genellikle yaşlanma süreciyle ilişkilidir; beyinde sinir hücrelerinin ölümüne ve bağlantı kaybına neden olur. Son zamanlarda, derin öğrenme modelleri kullanılarak Alzheimer hastalığının teşhisi konusunda umut verici sonuçlar elde edilmiş ve erken teşhis için çeşitli yaklaşımlar önerilmiştir. Ancak, özellikle nadir durumları içeren sağlık veri setlerindeki dengesizlik, model eğitimi sırasında performans kayıplarına ve yanıltıcı sonuçlara yol açabilir. Bu çalışma, bu dengesizlik sorunlarına odaklanarak, Alzheimer MRI veri seti için farklı dengeleme yöntemlerinin etkinliğini değerlendirmektedir. Bu bağlamda, özel bir model kullanılarak SMOTE, ADASYN ve Ağırlık Dengesi yöntemlerinin performansı karşılaştırılmaktadır.

*Sorumlu yazar (Corresponding Author): Yahya DOĞAN, *yahyadogan@siirt.edu.tr*

Deneysel sonuçlar, orijinal dengesiz veri setine kıyasla Ağırlık Dengesi yönteminin doğruluk, hassasiyet, geri çağrı ve F1 skoru açısından daha üstün olduğunu göstermektedir. SMOTE ve ADASYN, çeşitli metriklerde iyileşme göstermesine rağmen, Ağırlık Dengesi yöntemine kıyasla daha düşük performansa sahip oldukları gözlemlenmiştir. Bu çalışma, Alzheimer hastalığı sınıflandırmasında derin öğrenme modellerinin doğruluğunu artırmak için veri dengeleme yöntemlerinin seçimine katkıda bulunur ve sağlık veri setlerinde sınıf dengesizliğinin ele alınmasının önemini vurgular.

Anahtar Kelimeler: Derin öğrenme, Evrişimsel sinir ağları, SMOTE, ADASYN, Ağırlık dengeleme

1. INTRODUCTION

Alzheimer's disease, a prevalent neurological disorder that impacts approximately 50 million individuals globally, presents a significant challenge in the field of healthcare [1]. As this neurodegenerative disease progresses, it poses a serious threat to an individual's general health, potentially leading to death in the event of complete brain failure. Because of the broad loss of nerve cells across the brain, Alzheimer's has a far-reaching impact, extending to basic skills such as writing, speaking, and reading. Notably, people in the cognitive stages of Alzheimer's disease may have difficulty identifying their family members.

The insidious nature of Alzheimer's disease, distinguished by its gradual onset of symptoms, makes accurate and early diagnosis difficult [2]. The importance of early-stage identification, on the other hand, cannot be stressed, as it allows for rapid intervention and therapy, ultimately contributing to a better prognosis for people suffering from this complicated neurodegenerative disorder [3]. In this context, developing efficient diagnostic methods is critical to improving our ability to combat Alzheimer's disease and lessen its devastating effects on people and society as a whole.

Deep learning algorithms have seen substantial success in a variety of fields [4-7]. Consequently, the importance of deep learning-based approaches in the diagnosis of Alzheimer's disease has been rapidly increasing [8-11]. Various methods have been presented in this field to assist clinicians in making educated medical decisions as diagnostic aids for Alzheimer's disease. Lu et. al. [12] introduced an innovative multimodal deep neural network employing a multistage technique for the detection of dementia. Their method demonstrated notable success, achieving an accuracy of 82.4% in predicting mild cognitive impairment (MCI) and identifying individuals who later developed Alzheimer's disease within three years. The model exhibited a notable sensitivity of 94.23% in Alzheimer's disease detection and achieved an accuracy of 86.3% for the non-demented class. Ahmed et. al. [13] proposed an ensemble CNN model for Alzheimer's disease (AD) diagnosis that used a feature extractor and the Softmax classifier. The model, designed to avoid overfitting, performed well, obtaining an accuracy of 90.05% by utilizing MRI images centered on the left and right hippocampal sections. Liu et. al. [14] utilized siamese neural networks to assess whole-brain volumetric asymmetry. They used the MRI cloud approach to produce low-dimensional descriptors for designated atlas brain structures. They employed a unique non-linear kernel method to normalize features, eliminating batch effects across different datasets and populations. Using the ADNI dataset, the networks achieved a balanced accuracy of 92.72% in the classification of MCI and Alzheimer's disease. Sarraf et. al. [15] suggested a deep learning pipeline for feature categorization that focused on processes that don't change with scale or shift and included a CNN model trained on a large dataset. The model performed well, with accuracy rates of 94.32% for functional MRI and 97.88% for MRI images.

The analysis of datasets in the field of healthcare typically begins with the challenges encountered in the process of collecting samples related to specific health conditions or diseases. One of these challenges arises from the rarity of certain health conditions or the limited number of samples belonging to specific classes, leading to an imbalance in the datasets. The collection of samples associated with rarely occurring diseases or specific health conditions can adversely affect the effectiveness of analysis and classification models. This imbalance can lead to performance losses and misleading outcomes during model training. This study focuses on addressing the imbalance issues in healthcare datasets and investigates the performance of methods developed to overcome this challenge.

In the case of Alzheimer's disease diagnosis with deep learning algorithms, imbalanced datasets are a common issue. This occurs when one class has significantly more examples than the other. For instance, in Alzheimer's disease, the number of non-diseased individuals typically far exceeds that of diseased individuals. This imbalance negatively impacts the performance of classification models. In imbalanced datasets, the model tends to overfit the majority class and fails to learn accurately from the minority class. Consequently, it correctly classifies majority class examples but performs poorly on minority class examples. This leads to significant performance issues, especially when the minority class is critically important. In Alzheimer's disease, misclassification of the minority class can have serious consequences, such as missed opportunities for early intervention and treatment. Although overall accuracy might appear high, these metrics can be misleading for the minority class. In this context, the performance of dataset balancing methods is analyzed using the Alzheimer's MRI dataset.

2. MATERIALS AND METHODS

This section covers the dataset, model, strategies for dealing with data imbalances, training details, and metrics.

In this study, the Alzheimer's MRI Dataset [16] from the Kaggle website was used, which included four separate classes: non-demented, very mild, mild, and moderate. It is important to highlight that the dataset is unbalanced. The class distributions are as follows: Non-demented (3200 samples), very mild (2240 samples), moderate (64 samples), and mild (896 samples). Figure 1 shows a histogram indicating the distribution of each class. This dataset comprises a total of 6400 MRI images representing various levels of dementia.

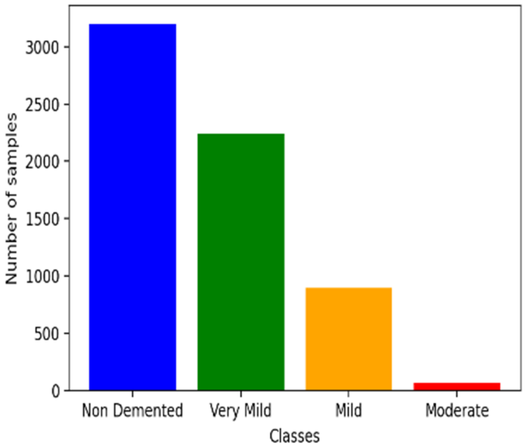


Figure 1. Class distribution of the Alzheimer's MRI dataset

The dataset was partitioned into three independent subsets during the training process-training, testing, and validation sets-to properly evaluate the model's performance. Specifically, 85% of the dataset has been set aside for training, with the remaining 15% set aside for testing. Furthermore,15% of the training dataset has been put aside to serve as the validation set to assess the model's generalization abilities. Figure 2 shows several random samples from the dataset

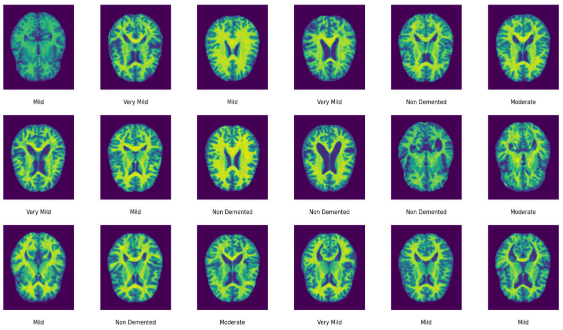


Figure 2. Random samples from the Alzheimer's MRI dataset

A. Model

To assess the performance of proposed interventions for the data imbalance problem, a custom model was created. The primary objective of this model is to classify a given input image and generate probability distributions for four different classes as outputs. The model incorporates convolutional layers for feature extraction and complexity reduction, activation functions to introduce non-linearity, pooling layers to decrease the size of feature maps, batch normalization layers to normalize inputs for each layer within the network, and dense layers for classification purposes. In Figure 3, details of each layer used in the model, such as the number of kernels, kernel size, input, and output dimensions, are provided.

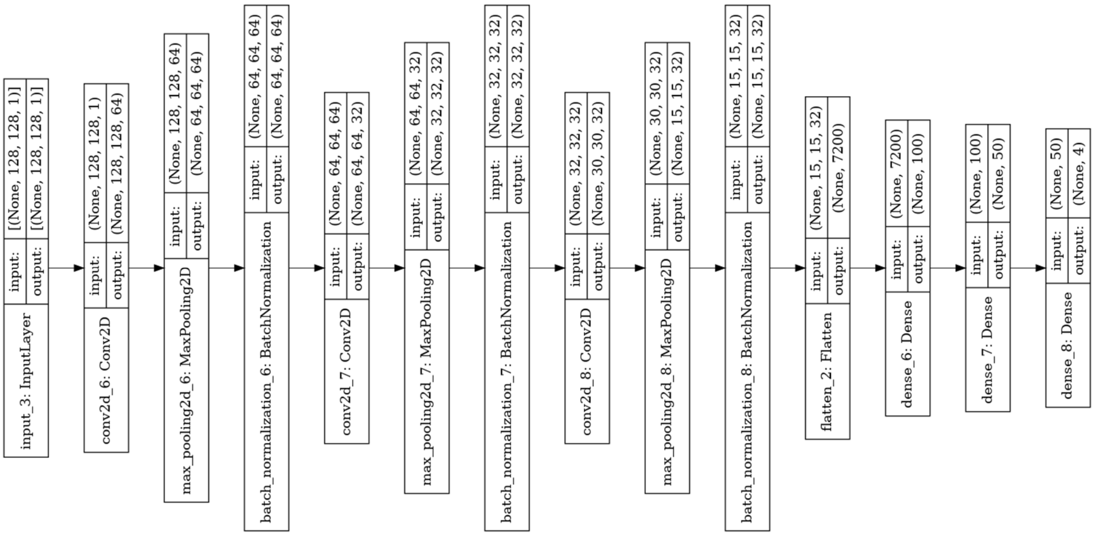

input_3: InputLayer
input: [(None, 128, 128, 1)]
output: [(None, 128, 128, 1)]
conv2d_6: Conv2D
input: (None, 128, 128, 1)
output: (None, 128, 128, 64)
max_pooling2d_6: MaxPooling2D
input: (None, 128, 128, 64)
output: (None, 64, 64, 64)
batch_normalization_6: BatchNormalization
input: (None, 64, 64, 64)
output: (None, 64, 64, 64)
conv2d_7: Conv2D
input: (None, 64, 64, 64)
output: (None, 64, 64, 32)
max_pooling2d_7: MaxPooling2D
input: (None, 64, 64, 32)
output: (None, 32, 32, 32)
batch_normalization_7: BatchNormalization
input: (None, 32, 32, 32)
output: (None, 32, 32, 32)
conv2d_8: Conv2D
input: (None, 32, 32, 32)
output: (None, 30, 30, 32)
max_pooling2d_8: MaxPooling2D
input: (None, 30, 30, 32)
output: (None, 15, 15, 32)
batch_normalization_8: BatchNormalization
input: (None, 15, 15, 32)
output: (None, 15, 15, 32)
flatten_2: Flatten
input: (None, 15, 15, 32)
output: (None, 7200)
dense_6: Dense
input: (None, 7200)
output: (None, 100)
dense_7: Dense
input: (None, 100)
output: (None, 50)
dense_8: Dense
input: (None, 50)
output: (None, 4)

Figure 3. A custom model created for comparing various approaches to address the issue of data imbalance

B. Methods for Addressing Data Imbalance

In this section, we discuss dataset balancing methods for imbalanced classification problems, where there is a skewed distribution of classes in the dataset and one class (usually the minority class) has significantly fewer examples than the others.

1. SMOTE (Synthetic Minority Over-Sampling Technique) [17]: This method draws inspiration from a technique used in handwritten character recognition, aiming to produce synthetic examples for the minority class. Rather than resorting to replacement over-sampling, a more specific approach is introduced by producing synthetic examples in the feature space as opposed to the data space.

The minority class is the focus of this method, and synthetic examples are constructed by extending along line segments linking any or all of its k nearest neighbours. The number of neighbours chosen at random from the k nearest neighbours is determined by the degree of oversampling. The current implementation makes use of information from the five closest neighbours. For example, if a 200% over-sampling is desired, two neighbours are chosen at random from the five closest neighbours, and a synthetic example is constructed in each direction.

Creating synthetic examples entails calculating the difference between the current example's feature vector and its nearest neighbor. This difference is then multiplied by a number between 0 and 1, and the result is added to the feature vector. As a result, a point is generated randomly within the line segment connecting two specific features. This novel method efficiently broadens the deciding zone of the minority class, encouraging broader representation.

SMOTE offers several advantages and disadvantages for addressing class imbalances in datasets. Advantages include improved minority class representation by generating synthetic examples, enhanced model performance in terms of accuracy, recall, and F1 score by providing a balanced training dataset and reducing overfitting by creating varied samples instead of duplicating minority class samples. Additionally, SMOTE helps establish better decision boundaries by broadening the minority class's decision zone and applies to various classification algorithms, making it a versatile tool.

Disadvantages include the potential introduction of synthetic noise into the dataset, which can negatively impact model performance if the generated samples are not representative of the true data distribution. The method also increases computational complexity as synthetic samples are generated and the k-nearest neighbors are determined, particularly for large datasets. There is a risk of overgeneralization, where the classifier may become too lenient in distinguishing between classes, reducing specificity. Furthermore, SMOTE's effectiveness is sensitive to parameter settings, such as the number of nearest neighbors (k) and the degree of over-sampling; poorly chosen parameters can adversely affect performance. In high-dimensional feature spaces, the nearest neighbor search and synthetic sample generation can become less effective, potentially leading to suboptimal results.

2. ADASYN (Adaptive Synthetic Sampling) [18]: In this method, an adaptive approach, inspired by recently successful synthetic methods such as SMOTE [17], SMOTEBoost [19], and DataBoostIM [20], is proposed to make learning from imbalanced datasets. The dual goal is to decrease bias while also enabling adaptive learning. The fundamental concept underlying the ADASYN algorithm involves employing a density distribution as a criterion to autonomously determine the number of synthetic samples required for each minority data instance. The degree of class imbalance is initially estimated using Equation 1 as follows:

$$d = m_s / m_l \tag{1}$$

Where $d \in (0, 1]$ defines m_s and m_l as the quantities of minority and majority class instances, respectively. If d is less than a predefined threshold for the maximum tolerable degree of class imbalance, Equation 2 determines the required amount of synthetic data samples for the minority class. This ratio $d \in (0, 1]$ indicates the extent of imbalance in the dataset. A lower value of d signifies a higher degree of imbalance.

$$G = (m_l - m_s)\ x\ \beta \tag{2}$$

The total number of synthetic samples G needed for the minority class is determined by the difference between the majority class instances m_l and the minority class instances m_s, multiplied by the balance level parameter β. This ensures that the dataset moves towards a balanced state as defined by β. The parameter $\beta \in [0, 1]$ indicates the targeted balance level following synthetic data generation. A value of $\beta = 1$ signifies the creation of a completely balanced dataset after the augmentation process. For each instance, x_i in the minority class, the K nearest neighbors are identified based on Euclidean distance in an n-dimensional space, and the r_i ratio is calculated as follows:

$$r_i = \frac{\Delta_i}{K}, \quad i = 1, 2, 3, \ldots, m_s \tag{3}$$

Where Δ_i represents the count of instances in the K nearest neighbors of x_i that are from the majority class, resulting in $r_i \in [0, 1]$. This ratio helps in identifying how challenging it is for the model to classify the minority instance correctly. r_i is then normalized as follows:

$$\hat{r}_i = r_i / \sum_{i=1}^{m_s} r_i \tag{4}$$

Where $\hat{r}_i$ represents a density distribution. This ensures that the synthetic samples are generated in proportion to the difficulty of the minority instances. The computation of the quantity of synthetic data samples to generate for each minority instance x_i is determined as follows:

$$g_i = \hat{r}_i\ x\ G \tag{5}$$

Where G denotes the overall quantity of synthetic data instances needed for the minority class, as specified in Equation (2). This adaptive approach ensures that more synthetic samples are generated for the harder-to-learn minority instances. For each minority class data instance x_i, a loop from 1 to g_i is created, and synthetic data samples are generated using the following equation.

$$s_i = x_i + (x_{zi} - x_i)\, x\, \lambda \qquad (6)$$

A randomly selected minority data instance, x_{zi}, is chosen from the K nearest neighbors of data x_i. Here, $(x_{zi} - x_i)$ is the vector of distinction in an n-dimensional space, and λ is a random number: λ ∈ [0,1]. Physically, it is a metric that evaluates how weights are distributed among various instances within the minority class, considering their respective difficulty levels in learning. The dataset obtained after applying ADASYN not only achieves a balanced representation of the data distribution based on the specified balance level (determined by the β coefficient) but also directs the learning algorithm to focus on these particularly challenging instances. This is a significant distinction, particularly when compared to the SMOTE, where an equal number of synthetic samples are produced for each minority data instance.

ADASYN offers several advantages and disadvantages for addressing class imbalance in datasets. Advantages involve adaptive sample creation, which creates more synthetic examples for harder-to-learn instances, allowing the model to focus on challenging cases. This adaptive approach can improve overall classifier performance, particularly in terms of recall for the minority class. ADASYN provides dynamic balancing by adjusting the number of synthetic samples based on the density distribution of the minority class, resulting in a more balanced and representative dataset. Additionally, by generating synthetic samples based on the local data distribution, ADASYN helps reduce the risk of overfitting compared to methods that simply duplicate minority class samples. Disadvantages include increased computational complexity due to the need to calculate nearest neighbors and density distributions, especially for large datasets. The method can potentially introduce noise if the synthetic examples do not accurately represent the underlying data distribution. The performance of ADASYN is sensitive to parameters such as the number of nearest neighbors (K) and the balance level (β), and improper parameter settings can lead to suboptimal results. In high-dimensional spaces, the nearest neighbor search and the generation of synthetic samples can become less effective, leading to poor model performance. Furthermore, the adaptive nature of ADASYN can make it more complex to implement and tune compared to simpler over-sampling methods.

3. Weight balancing [21]: While most deep learning algorithms tend to struggle with biased class data, the effectiveness of these models can be significantly enhanced by adapting existing training algorithms to accommodate the skewed distribution of classes. This adaptation involves assigning distinct weights to both majority and minority classes, thereby influencing the classification dynamics during the training phase. The primary objective is to penalize misclassifications made by the minority class by augmenting its class weight, while simultaneously diminishing the weight of the majority class.

One commonly employed method in this context is weight balancing, frequently utilized in classification models. This method entails the assignment of varying weights to different classes, aiming to increase the model's sensitivity to the minority class. Many classification algorithms offer the flexibility to assign class weights during training, with higher weights allocated to the minority class. This strategic weighting makes misclassifications of minority class instances more impactful in terms of the overall loss function, motivating the model to prioritize and improve predictions for the minority class. Weighting is applied to different classes in a dataset, taking into account the distribution of example counts across the classes, as illustrated below.

$$w_i = \frac{Ns}{Nc\, x\, Ns_i} \qquad (7)$$

Where w_i represents the weight assigned to each class, Ns denotes the total sample count, Nc refers to the total count of unique classes within the target variable, Ns_i represents the total number of rows associated with the respective class i. By dividing the total number of samples by the product of the number of classes and the samples corresponding to each class, the formula dynamically calculates weights that reflect the relative proportions of each class in the dataset. This ensures a more nuanced and balanced consideration of different classes during the model training process, contributing to improved performance, especially in scenarios with imbalanced class distributions.

Weight balancing offers several advantages and disadvantages in addressing class imbalance in datasets. One significant advantage is its ability to enhance model sensitivity by increasing the weights of the minority class, thus improving recall and precision for these instances. Moreover, it is a versatile technique that can be easily applied to various machine learning and deep learning frameworks, making it flexible and adaptable. Additionally, weight balancing contributes to overall performance enhancement by reducing bias towards the majority class, thereby improving metrics such as F1-score and AUC. It can also be seamlessly integrated into different classification algorithms, further enhancing its utility. From an implementation standpoint, weight balancing is relatively simple, often requiring only minor adjustments to the loss function or training process.

However, weight balancing is not without its drawbacks. One potential issue is the risk of overcompensation, where setting weights too high for the minority class can lead to overfitting and poor generalization of new data. Furthermore, altering weights during training might result in computational costs, especially for massive datasets with many classes. The effectiveness of weight balancing is heavily dependent on accurately estimating class distribution, and incorrect weight assignments can result in suboptimal performance. Tuning weight values to find the optimal balance requires careful validation, which can be time-consuming and computationally expensive. Lastly, weight balancing may have a limited impact on severely imbalanced datasets, necessitating the use of additional techniques such as synthetic data generation for better results.

C. Training Details

The model was trained for 20 epochs from scratch, with a categorical loss function used throughout the training phase. Because of its known success with large datasets and complex models, the Adam algorithm [22] was used as the optimizer. During training, validation accuracy was continually assessed to evaluate model performance and minimize overfitting. In the convolutional and dense layers, ReLU activation functions were utilized, whereas the softmax activation function was utilized in the final fully connected layer. For each data imbalance approach, the model was trained from scratch.

D. Metrics

The confusion matrix is a commonly used set of metrics for determining how effectively classification models perform. The confusion matrix encompasses four distinct concepts: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). A confusion matrix is typically represented in the following tabular format:

Table 1. The confusion matrix breaks down predictions into four categories: TP when the actual class is positive and the model correctly predicts it as positive; FP when the actual class is negative, but the model incorrectly predicts it as positive; TN when the actual class is negative, and the model correctly predicts it as negative; FN when the actual class is positive, but the model incorrectly predicts it as negative

	Positive	Negative
Predicted positive	TN	FP
Predicted negative	FN	TP

To evaluate method performance through the confusion matrix, four metrics were used: accuracy, precision, recall, and F1 score. Accuracy assesses the proportion of correctly predicted instances by a model, serving as a comprehensive measure to evaluate the overall effectiveness of a classification model.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

The precision metric assesses the accuracy of a model's positive predictions by determining the proportion of correctly identified positive instances. This metric specifically focuses on minimizing the occurrence of false positive predictions in a classification model.

$$Precision = \frac{TP}{TP+FP} \tag{9}$$

The recall metric assesses a model's ability to accurately identify positive instances, providing the percentage of real positive values that were correctly predicted. The goal of this metric is to reduce the number of incorrect negative predictions in a classification model.

$$Recall\ = \frac{TP}{TP+FN} \tag{10}$$

The F1 score is a metric that represents the harmonic mean of precision and recall metrics, offering a balanced measure that considers both false positive and false negative predictions generated by a classification model.

$$F1\ Score = 2\ x \frac{Precision\ x\ Recall}{Precision+Recall} \tag{11}$$

3. EXPERIMENT AND RESULTS

This section presents a quantitative comparison of the SMOTE, ADASYN, and Weight balancing methods for the Alzheimer MRI dataset with unbalanced sample counts among classes. In this context, we first trained a custom model from scratch using the existing dataset and then analyzed its performance. The classification report of the model is presented in Table 2.

Table 2. Model performance without dataset balancing

	Precision	Recall	F1-score
0	0.98	0.97	0.98
1	0.94	0.96	0.95
2	0.99	0.95	0.97
3	1.0	0.83	0.91
Accuracy			0.97
Macro Avg	0.98	0.93	0.95
Weighted Avg	0.97	0.97	0.97

In the absence of applying a dataset balancing method, an accuracy value of 97% was obtained. Upon analyzing the overall classification performance through macro avg, encompassing precision, recall, and f1-score metrics, scores of 98%, 93%, and 95% were respectively achieved. Macro avg provides a comprehensive overview, particularly beneficial in evaluating performance metrics collectively in multi-class classification problems. It takes into account the imbalance among classes by treating each class's contribution equally. It computes and subsequently averages performance metrics for each class, considering their contributions. This approach ensures a fair evaluation of overall performance, irrespective of significant variations in performance across classes. When evaluated for specific classes, Class 3, with a limited number of instances, shows significantly low recall and F1-score metrics. Figure 4 depicts the validation loss and accuracy graphs for the relevant model.

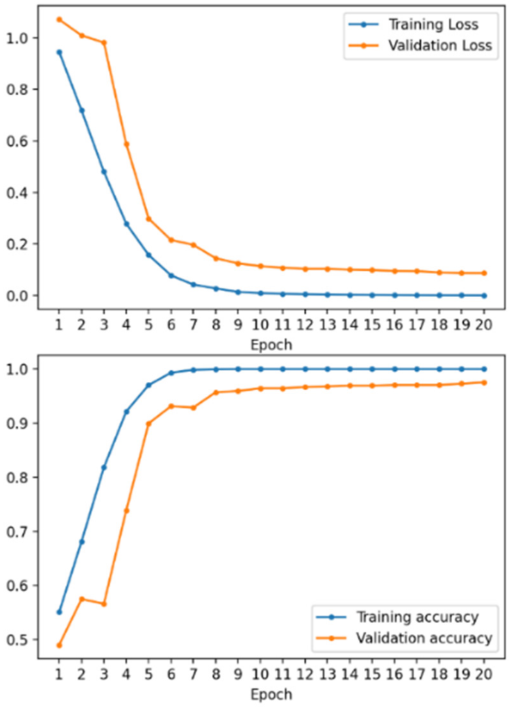
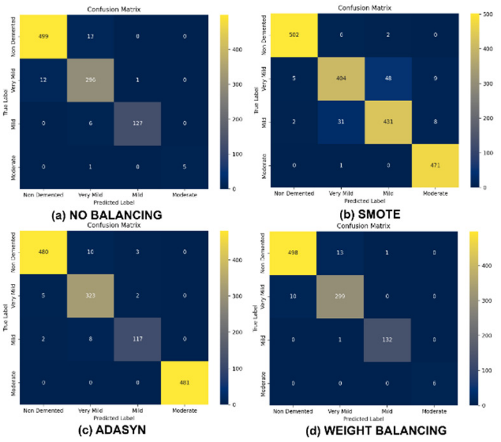


Figure 4. Validation loss and accuracy graphs of the model trained without applying data balancing.

Figure 5 (a) presents the confusion matrix for the unbalanced dataset. Notably, 13 instances from the non-demented class are misclassified as very mild, while the remaining 499 instances are correctly classified. Similarly, in the very-mild class, 12 instances are misclassified as non-demented, 1 as mild, and the remaining 296 instances are correctly classified. For the mild class, 6 instances are misclassified as very-mild, while the remaining 127 instances are correctly classified. Lastly, in the moderate class, 1 instance is misclassified as very mild, and the remaining 5 instances are correctly classified.

In the second stage, the model performance was examined by applying the data balancing process using the SMOTE method. In this approach, the sample count for each class was designed to be the same as the class with the highest number of instances. Accordingly, the imbalance was addressed, ensuring that the sample count for each class was adjusted to 3200.



(a) NO BALANCING

(b) SMOTE

(c) ADASYN

(d) WEIGHT BALANCING

Figure 5. Comparison of confusion matrix results: (a) Unbalanced, (b) SMOTE balanced, (c) ADASYN balanced, and (d) Weight balanced.

Table 3 presents the performance report for the relevant method. When the SMOTE method is used for balancing, a decrease in model performance is observed. The accuracy value has decreased from 97% to 94% compared to the original dataset. Similarly, precision and f1-score decrease, while the recall value increases by 1%.

Table 3. Model performance when the SMOTE data balancing method is applied.

	Precision	**Recall**	**F1-score**
0	0.99	0.98	0.99
1	0.91	0.87	0.89
2	0.90	0.91	0.90
3	0.97	1.0	0.98
Accuracy			0.94
Macro Avg	0.94	0.94	0.94
Weighted Avg	0.94	0.94	0.94

Figure 6 shows the validation and loss graphs for the model trained using the relevant approach. In addition, Figure 5(b) shows the confusion matrix for the model trained using the SMOTE approach. In the non-demented class (510 examples), 6 are very mild, 1 is mild, and the rest are correctly classified. For the very mild class (466 examples), 5 are non-demented, 48 are mild, 9 are moderate, and the remaining are correctly classified. In the mild class (472 examples), 2 are non-demented, 31 are very mild, 8 are moderate, and the rest are correctly classified. In the moderate class (472 examples), 1 is misclassified as very mild, and the rest are correctly classified.

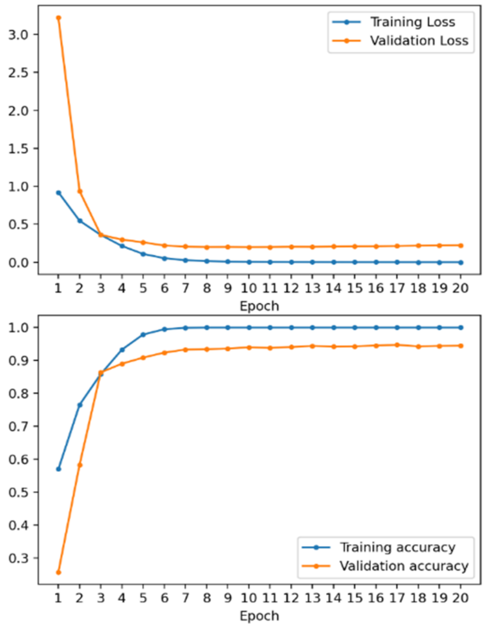


Figure 6. Validation loss and accuracy graphs of the model trained to apply the SMOTE data balancing method.

In the third stage, the model's performance was assessed using the ADASYN method, and Table 4 displays the corresponding performance report. Results indicate superior performance with ADASYN compared to the original and SMOTE methods for the Alzheimer's MRI Dataset. For the class with the fewest examples, i.e., 3, the original model had a recall and f1 scores of 83% and 91%, respectively. These scores were 100% and 98% in the SMOTE method, respectively, while the ADASYN method achieved 100% performance in both metrics. Overall averages show 97% accuracy in the original dataset, %94 with SMOTE, and 98% with ADASYN.

Table 4. Model performance when the ADASYN data balancing method is applied.

	Precision	Recall	F1-score
0	0.99	0.97	0.98
1	0.95	0.98	0.96
2	0.96	0.92	0.94
3	1.0	1.0	1.0
Accuracy			0.98
Macro Avg	0.97	0.97	0.97
Weighted Avg	0.98	0.98	0.98

Figure 5(c) presents the confusion matrix derived from the training process employing the ADASYN method. Within the non-demented class, which included 493 examples, 10 instances were classified as very mild, 3 as mild, and the remaining were accurately classified. In the very mild class, which consisted of 330 examples, 5 were designated as non-demented, 3 as mild, and the remainder were correctly classified. The mild class, which consisted of 127 examples, saw two cases classified as non-demented, eight as very mild, and the remainder correctly classified. Ultimately, within the moderate class, comprising 481 examples, all instances were accurately classified. Figure 7 illustrates the validation and accuracy graphs of the model trained using the ADASYN data balancing method.

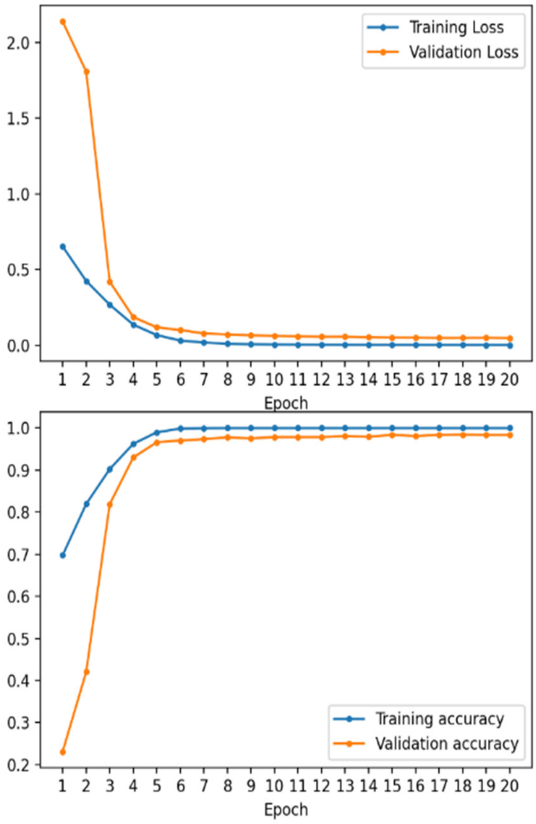


Figure 7. Validation loss and accuracy graphs of the model trained to apply the ADASYN data balancing method.

Finally, the model's performance was assessed using the Weight-balancing method. In this approach, the dataset is not augmented; instead, weights are assigned based on the distribution of examples in the dataset. The goal is to increase the weight of the minority class to enhance the model's attention to it. In this context, considering the number of examples in the dataset, weight values of 0.5, 0.71, 1.79, and 25.0 were assigned to the non-demented, very mild, mild, and moderate classes, respectively. Table 5 summarizes the results from training the model with this method.

Table 5. Model performance when the Weight balancing method is applied.

	Precision	**Recall**	**F1-score**
0	0.98	0.97	0.98
1	0.96	0.97	0.96
2	0.99	0.99	0.99
3	1.0	1.0	1.0
Accuracy			0.97
Macro Avg	0.98	0.98	0.98
Weighted Avg	0.97	0.97	0.97

Upon examining Table 5, it is observed that the highest scores are obtained, particularly for the minority class, i.e., moderate. Looking at the macro average scores, the model achieved the highest performance scores, reaching 98% for precision, recall, and f1-score. When compared to the model trained with the original dataset, there is a 5% increase in the recall metric and a 3% increase in the f1-score metric. In Figure 5(d), the confusion matrix obtained when the model is trained using the weight balancing method is provided. The results show that, out of 512 examples in the non-demented class, 13 were classified as very mild, 1 as mild, and the rest were correctly classified. In the very mild class, with 309 examples, 10 were misclassified as non-demented, while the others were correctly classified. In the mild class with 133 examples, only 1 example was incorrectly classified as very mild. Finally, in the moderate class with six examples, all were correctly classified. Figure 8 displays the validation loss and accuracy graphs obtained when the model is trained using the weight balancing method.

Ultimately, the test dataset was separated before applying data balancing methods, and the performance of the methods was evaluated in this manner. Table 7 reveals that similar to previous experimental results, the weight-balancing method outperforms the others. The SMOTE approach produces worse results than the model trained on the original dataset. Similarly, the ADASYN method also yields better results than the original model.

Table 6. Performance comparison of data balancing methods on test dataset

	Accuracy	Precision	Recall	F1 score
SMOTE	94.17	94.05	94.03	94.02
ORJINAL	96.56	93.01	97.63	95.12
ADASYN	**97.90**	96.84	97.30	97.05
Weight balancing	97.40	**98.32**	**98.20**	**98.26**

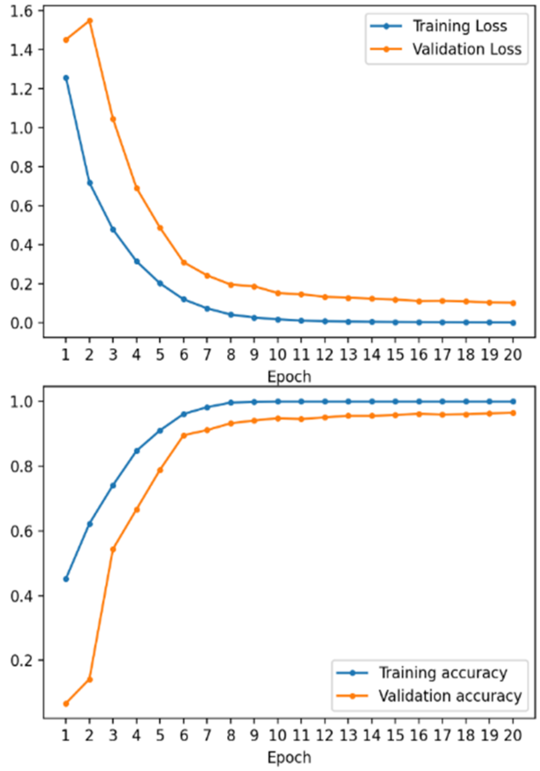


Figure 8. Validation loss and accuracy graphs of the model trained to apply the Weight balancing method.

4. CONCLUSION

This study aimed to address the challenges posed by imbalanced datasets in Alzheimer's disease classification, focusing on the effectiveness of three data balancing methods: SMOTE, ADASYN, and Weight Balancing. The experiments were conducted using the Alzheimer's MRI dataset, and a custom deep-learning model was used for evaluation. The results indicate that, compared to the original imbalanced dataset, the Weight Balancing method consistently outperforms in terms of accuracy, precision, recall, and F1 score. The method assigns weights based on class distribution, enabling the model to pay more attention to the minority class, which is particularly beneficial in the context of imbalanced health datasets. While SMOTE and ADASYN methods improve various metrics, they are considered inferior to the Weight Balancing method. The ADASYN method, in particular, demonstrated superior performance, achieving the highest scores for precision, recall, and F1 score, especially for the minority class, i.e., moderate. The study emphasizes the critical role of addressing class imbalances in health datasets for accurate and reliable model training. It contributes valuable insights into selecting data balancing methods to enhance the accuracy of deep learning models in Alzheimer's disease classification. The Weight Balancing method, with its ability to adapt class weights during training, stands out as a robust approach for improving model performance in scenarios with imbalanced class distributions. In future work, further exploration of different data balancing methods and validation on diverse datasets could provide additional perspectives on optimizing deep learning models for Alzheimer's disease classification.

5. REFERENCES

1. Nawaz, H., Maqsood, M., Afzal, S., Aadil, F., Mehmood, I., Rho, S., 2021. A Deep Feature-Based Real-Time System for Alzheimer Disease Stage Detection. Multimedia Tools and Applications, 80, 35789-35807.
2. Aditya Shastry, K., Sanjay, H.A., 2023. Artificial Intelligence Techniques for the Effective Diagnosis of Alzheimer’s Disease: A Review. Multimedia Tools and Applications, 83(13), 40057-40092.
3. Yao, Z., Mao, W., Yuan, Y., Shi, Z., Zhu, G., Zhang, W., Wang, Z., Zhang, G., 2023. Fuzzy-VGG: A Fast Deep Learning Method for Predicting the Staging of Alzheimer's Disease Based on Brain MRI. Information Sciences, 642, 119129.
4. Özdemir, C., 2023. Designing Effective Models for COVID-19 Diagnosis through Transfer Learning and Interlayer Visualization. Balkan Journal of Electrical and Computer Engineering, 11(4), 340-345.
5. Sivari, E., Civelek, Z., Sahin, S., 2024. Determination and Classification of Fetal Sex on Ultrasound Images with Deep Learning. Expert Systems with Applications, 240, 122508.
6. Kılıç, Ş., Doğan, Y., 2023. Deep Learning Based Gender Identification Using ear Images. Traitement du Signal, 40(4), 1629-1639.
7. Ozdemir, C., 2023. Classification of Brain Tumors from MR Images Using a New CNN Architecture. Traitement du Signal, 40(2), 611-618.
8. Assmi, A., Elhabyb, K., Benba, A., Jilbab, A., 2024. Alzheimer’s Disease Classification: A Comprehensive Study. Multimedia Tools and Applications, 1-24.
9. Mujahid, M., Rehman, A., Alam, T., Alamri, F. S., Fati, S. M., Saba, T., 2023. An Efficient Ensemble Approach for Alzheimer’s Disease Detection Using an Adaptive Synthetic Technique and Deep Learning. Diagnostics, 13(15), 2489.
10. Borkar, P., Wankhede, V.A., Mane, D.T., Limkar, S., Ramesh, J.V.N., Ajani, S.N., 2023. Deep Learning and Image Processing-Based Early Detection of Alzheimer Disease in Cognitively Normal Individuals. Soft Computing, 1-23.
11. Thangavel, P., Natarajan, Y., Preethaa, K.S., 2023. EAD-DNN: Early Alzheimer's Disease Prediction Using Deep Neural Networks. Biomedical Signal Processing and Control, 86, 105215.
12. Lu, D., Popuri, K., Ding, G.W., Balachandar, R., Beg, M.F., 2018. Multimodal and Multiscale Deep Neural Networks for the Early Diagnosis of Alzheimer’s Disease Using Structural MR and FDG-PET Images. Scientific Reports, 8(1), 5697.
13. Ahmed, S., Choi, K.Y., Lee, J.J., Kim, B.C., Kwon, G.R., Lee, K.H., Jung, H.Y., 2019. Ensembles of Patch-Based Classifiers for Diagnosis of Alzheimer Diseases. IEEE Access, 7, 73373-73383.

14. Liu, C.F., Padhy, S., Ramachandran, S., Wang, V.X., Efimov, A., Bernal, A., Shi, L., Vaillant, M., Ratnanather, J.T., Faria, A.V., 2019. Using Deep Siamese Neural Networks for Detection of Brain Asymmetries Associated with Alzheimer's Disease and Mild Cognitive Impairment. Magnetic Resonance Imaging, 64, 190-199.

15. Sarraf, S., DeSouza, D.D., Anderson, J., Tofighi, G., 2016. DeepAD: Alzheimer's Disease Classification via Deep Convolutional Neural Networks Using MRI and fMRI. BioRxiv, 070441.

16. Alzheimer MRI Preprocessed Dataset, https://www.kaggle.com/datasets/sachinkumar413/alzheimer-mri-dataset, Access date: 08.01.2024.

17. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: Synthetic Minority Over-Sampling Technique. Journal of Artificial Intelligence Research, 16, 321-357.

18. He, H., Bai, Y., Garcia, E.A., Li, S., 2008. ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning. In 2008 IEEE International Joint Conference on Neural Networks, Hong Kong, China.

19. Chawla, N.V., Lazarevic, A., Hall, L.O., Bowyer, K.W., 2003. SMOTEBoost: Improving Prediction of the Minority Class in Boosting. In Knowledge Discovery in Databases: PKDD 2003: 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Cavtat-Dubrovnik, Croatia.

20. Guo, H., Viktor, H.L., 2004. Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach. ACM SigKDD Explorations Newsletter, 6(1), 30-39.

21. Du, M., Tatbul, N., Rivers, B., Gupta, A.K., Hu, L., Wang, W., Marcus, R., Zhou, S., Lee, I., Gottschlich, J., 2020. A Skew-Sensitive Evaluation Framework for Imbalanced Data Classification. arXiv preprint arXiv:2010.05995.

22. Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Pptimization. arXiv preprint arXiv:1412.6980.