International Journal of Biomedical Research & Health Sciences

IJBRHS | Volume 3 Issue 1

Robust Imputation and Classification of Parkinson's Disease Using a Questionnaire Dataset: A Novel Variational Autoencoder Approach

Elmira Mirzabeigi[1], Mohammad Javad Aghaei[2], Amir Hossein Karami[2], Sepehr Rezaee[2], Kourosh Parand[2,3]*

[1]Department of Applied Mathematics, Faculty of Mathematical Sciences, Tarbiat Modares University, Tehran, Iran
[2]Department of Computer Science and Data, Faculty of Mathematical Sciences, Shahid Beheshti University, Tehran,Iran
[3]Institute for Cognitive and Brain Sciences, Shahid Beheshti University, Tehran, Iran

***Corresponding author**: Kourosh Parand, Institute for Cognitive and Brain Sciences, Shahid Beheshti University, Tehran, Iran, E-mail: k_parand@sbu.ac.ir

Received date: 25-September-2025; **Editor assigned:** 29-September-2025; **Reviewed:** 03-October-2025; **Revised:** 07-October-2025; **Published:** 12-October-2025

Abstract

This study introduces a novel Fully Connected Variational Autoencoder (VICPute) to effectively address the challenges of missing data imputation and classification of Parkinson's disease (PD) using the Parkinson's Progression Markers Initiative (PPMI) dataset. The VICPute model, which integrates an encoder with dual decoders, refines imputation and classification processes through a pioneering architecture that employs computational blocks for enhanced feature extraction. This model distinguishes between healthy individuals and those with PD and robustly handles significant missing data issues within a complex dataset that includes 24,182 samples from 2,427 patients. Our approach optimizes training by applying advanced preprocessing techniques such as Multivariate Imputation by Chained Equations (MICE) and generating artificial missing data. The model's efficiency is underscored by a high f1-score of 0.87 in testing, highlighting the potential of deep learning techniques to enhance diagnostic accuracy and contribute to the broader understanding of PD progression.

***Keywords:* classification, imputation, Parkinson's disease, progression of disease, autoencoder, PPMI dataset**

Introduction

Parkinsons Disease (PD) represents one of the most common neurodegenerative disorders, affecting mil- lions worldwide. Characterized by the progressive loss of motor control, PD is accompanied by tremors, rigidity, and bradykinesia, alongside a spectrum of non-motor symptoms such as cognitive impairments and emotional disturbances [1, 2, 3, 4]. The etiology of PD involves the degeneration of dopamine-producing neurons in the brain, yet the precise mechanisms and biomarkers associated with disease progression remain under intensive study [5]. The diagnosis of PD has been enhanced through advanced imaging techniques and biomarkers [6]. Since the early 2000s, the focus has shifted towards finding non-invasive methods for early diagnosis [7, 8]. Mirelman et al. (2013) demonstrated how gait analysis could predict PD progression [9]. Later, the application of biomarkers in cerebrospinal fluid (CSF) and blood, as reviewed by Heinzel et al. (2019), provided new pathways for early-stage PD diagnosis [10, 11].

Citation: Parand K., Robust Imputation and Classification of Parkinson's Disease Using a Questionnaire Dataset: A Novel Variational Autoencoder Approach. Int. J. Bio. Res. Health. Science.2025;03(1):005.

Clinical data, encompassing neuroimaging, physiological, and genetic information, has been central to understanding and diagnosing PD [12, 13, 14, 15, 16, 17]. The Parkinsons Progression Markers Initiative (PPMI), launched in 2010, has been pivotal in collecting and analyzing such data [18]. Studies utilizing this rich dataset have explored the multifaceted nature of PD, examining everything from motor symptoms to cognitive decline [19]. Recent trends include integrating clinical data with artificial intelligence to predict disease progression and response to treatment. Furthermore, questionnaires have been instrumental in capturing patient-reported outcomes, which are crucial for clinical trials and routine management of PD. The non-motor symptoms questionnaire (NMSQuest), developed by Chaudhuri et al. (2006), has been widely used to assess the diverse symptoms of PD that are not related to motor function [20]. Recent research focuses on combining questionnaire data with sensor data to provide a more comprehensive assessment of patient health, thereby enhancing personalized treatment plans.

Machine learning methods are vital for advancing the understanding and treatment of neurodegenerative diseases by enabling the identification of patterns and predictors in vast datasets that are otherwise unmanageable by traditional analytical approaches [21]. Machine Learning (ML) techniques have greatly enhanced Parkinson's Disease (PD) diagnosis and monitoring, adeptly managing large datasets and effectively performing tasks like disease stage classification [22] and missing data imputation in longitudinal studies. Various ML methods, from traditional classifiers such as SVM and Random Forests to advanced deep learn- ing architectures, have been used to identify complex patterns undetectable through human analysis, thereby improving prediction accuracy. Notable research includes Sakar et al. (2019), who evaluated the efficacy of different ML techniques in PD classification, and Xiong and Lu (2020), who demonstrated the capability of deep learning in analyzing vocal data for PD detection [23, 24, 25]. Additionally, Mohammed et al. (2021) explored the use of variational autoencoders for imputing missing data, showcasing the versatility of ML in medical data analysis [16]. These studies lay a robust groundwork for further research into the use of ML in diagnosing neurodegenerative diseases, addressing both the inherent challenges and emerging opportunities. Our research aims to develop a Fully Connected Variational Denoising Autoencoder (FCVDAE) that ad- dresses the dual challenges of imputing missing data and accurately classifying individuals as either healthy or with Parkinson's Disease (PD). Utilizing the comprehensive dataset provided by the Parkinson's Progression Markers Initiative (PPMI), which encompasses a rich array of clinical [26, 27], behavioral, and biological data, our model employs a novel architecture inspired by Variational Autoencoder (VAE) principles. VAE model includes an advanced encoder-decoder setup with computational blocks that feature skip connections to enhance gradient flow and learning efficiency. By integrating robust data imputation with precise diagnostic classification, our approach improves the dataset's utility for advanced analyses deepening our understanding of PD, thereby facilitating the development of targeted therapeutic strategies. This innovative model architecture, equipped to handle the complexities and uncertainties inherent in clinical data, is trained using a composite loss function that optimizes imputation accuracy, classification performance, and conformity to the latent space distribution, ultimately paving the way for significant advancements in the management and study of Parkinsons Disease.

Preliminaries

In this section, we will describe the preliminary aspects of the model. The concept of a multilayer perceptron (MLP) model was first introduced in [28], which was created to handle nonlinearity in a model and is shown in the equation 1:

$$Z = \varphi(\theta_L \cdot \varphi(\theta_{L-1} \cdot \ldots \cdot \varphi(\theta_1 \cdot a + \beta_1) + \beta_2) + \ldots + \beta_L), \tag{1}$$

where θ_i and β_i are the weight and bias matrices for layer i, respectively. φ is an activation function, a is the input matrix, and Z is the output matrix.

In our work, we also utilize residual blocks introduced in [29]. As described in the original article, residual blocks improve model performance by allowing the input to bypass layers where no significant information is extracted. This mechanism ensures that the input is still available for subsequent layers. The formula for a residual block is given in equation 2:

$$\tau = a + \theta(a, \omega_i), \tag{2}$$

Where τ is the output, a is the input, and ω_i represents the weights associated with the layers within the residual block.

Another important concept is the Autoencoder (AE) []. The goal of an autoencoder, as shown in equation 3, is to reduce the dimensionality of the input and compress it into a latent space, which is a representation of the input. The latent space is then used to reconstruct the input as its original size. This helps to capture powerful features with a much smaller dimensionality.

$$\mu = \theta(a)$$
$$\hat{a} = \gamma(\mu), \tag{3}$$

Here, a is the input, θ is the encoder function, μ is the latent space, γ is the decoder function, and $\hat{a}$ is the reconstructed input. The closer $\hat{a}$ is to a, the richer the features in μ will be in terms of information extracted from the input.

Methodology

The complete process is illustrated in 1, which includes preprocessing, data preparation, and a presented model. In the following, each part of the process will be described in detail and finally, all results will be presented.

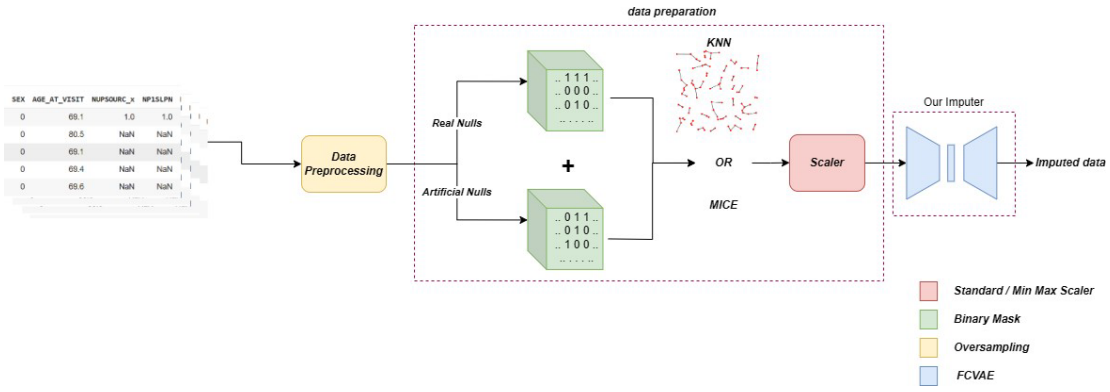

Figure 1. The general schema of the presented process: **Data Preprocessing:** Applying an oversampling for handling imbalanced dataset, **Mask Creation:** Creating a binary mask to represent known value elements, **Primary Imputation:** Imputing unknown values using a classic imputation method, **Scaler:** Scaling (e.g., normalization) as preprocessing, and **VICPute:** Applying the main imputation model.

1. Database Description

We have used the Parkinson's Progression Markers Initiative (PPMI) dataset for our purpose. This dataset contains various data collected from patients, including tables and images. The objective is to identify biomarkers that indicate disease progression using MRI, biological samples, and clinical and behavioral evaluations. Our work focuses on the questionnaire database, which includes different sections such as MDS-UPDRS (all parts), Physical Activity Scale for the Elderly - Household Activity, Modified Schwab

and England ADL, SCOPA-AUT, Clinical Cognitive Categorization, Geriatric Depression Scale (Short), Questionnaire for Impulsive-Compulsive Disorders (QUIP), State-Trait Anxiety Inventory, Benton Judgment of Line Orientation, Hopkins Verbal Learning Test, Letter-Number Sequencing (PD), Montreal Cognitive Assessment (MoCA), Semantic Fluency, Symbol Digit Modalities Test, Epworth Sleepiness Scale, and Features of REM Behavior Disorder. Furthermore, these tests have been performed over different periods. Therefore, it is necessary to exclude certain tests to maintain a logical number of missing values during the merging process. Some features were disregarded due to minimal overlap caused by different time steps. For merging, we included patients tested more than three times. In addition, time steps start from baseline BL (the first time they were tested) to V20/R20 (20 months after baseline). Although not all tests are available at all time steps (actual missing values), we consider them to make the model more robust. We had 24,182 samples from 2,427 patients available to start this investigation. The distribution of samples across time steps is shown in Figure 2.

According to Figure 2, the time steps include various types, such as V, BL, SC R, ST, U, and PW. BL refers to the baseline measurement taken when a participant is officially enrolled in the PPMI study. In some instances, SC may have preceded this, a screening measurement conducted to assess the participant's eligibility and gather static data, like demographics, typically around 60 days before the baseline. The V# indicate subsequent study visits during which tests were conducted and measurements recorded, while the R# represent remotely recorded measurements. Additionally, U# refers to unscheduled visits. The number of patients varies at different follow-up times. During certain time intervals, such as V01, V03, and V17, our participant count is notably low. Furthermore, we have a substantial number of patients in R01.

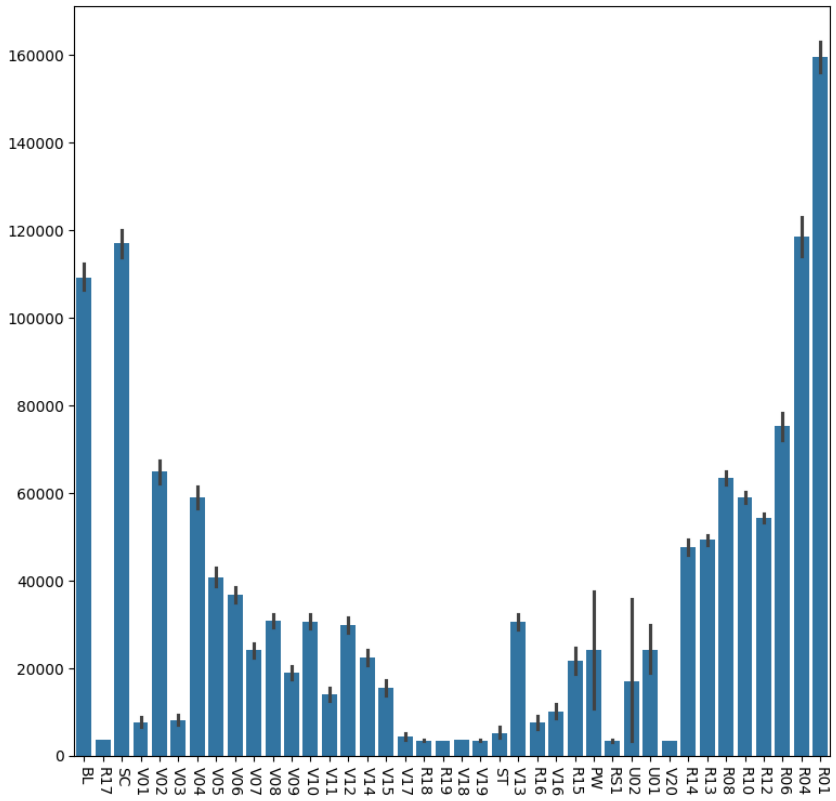

Figure 2. The Number of patients in different time steps

Figure 3 presents the histogram depicting the distribution of patients' ages. The x-axis indicates the ranges from 30 to 90 and the y-

axis represents the frequency, peaking at just under 1000 participants. The youngest and the oldest participants are 29 and 95, respectively. The mean and standard deviation of the age distribution are approximately 64.9 and 10.3, respectively. The distribution is roughly symmetric, forming a bell-shaped curve centered around the 65 to 70 age range. The number of participants decreases steadily on both sides of this peak, indicating fewer participants in the younger and older age groups.

The assessment of the patient's condition is essential for the classification task. We consider binary classification that groups individuals into two categories: having Parkinson's disease (represented as "Parkinson's Disease") and normal individuals (described as "Healthy"). Figure 4-5, it has been shown that the proportion of people with Parkinson's disease is much higher than healthy people. According to the data, the proportion of PDs is approximately three times higher than normal, leading to biased model results. There are approximately 4,000 confirmed cases of the disease, with over 12,000 individuals currently affected. Also, Table 1 compares the number of individuals with Parkinson's disease and those who are healthy, categorized by gender. There are more females with Parkinson's disease (7601) compared to males (4751). Conversely, the number of healthy individuals shows a different pattern, with males (2269) outnumbering females (1383). This suggests a notable difference in gender distribution between the two groups, with Parkinsons disease having more female participants, while the healthy group has more male participants. Hence, a gender disparity in the prevalence of Parkinson's disease is more common in females, while the healthy population has a higher proportion of males.

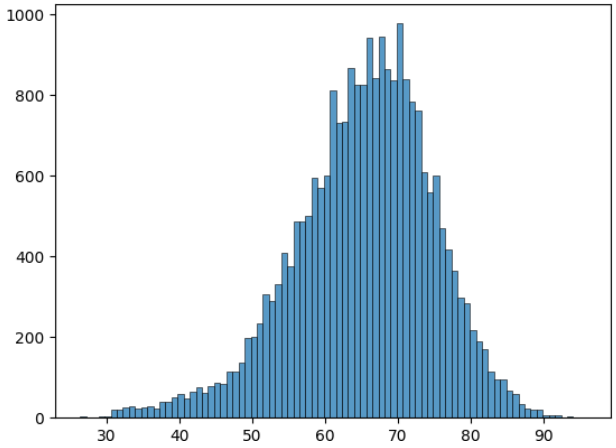

Figure 3. Histogram of participant's age

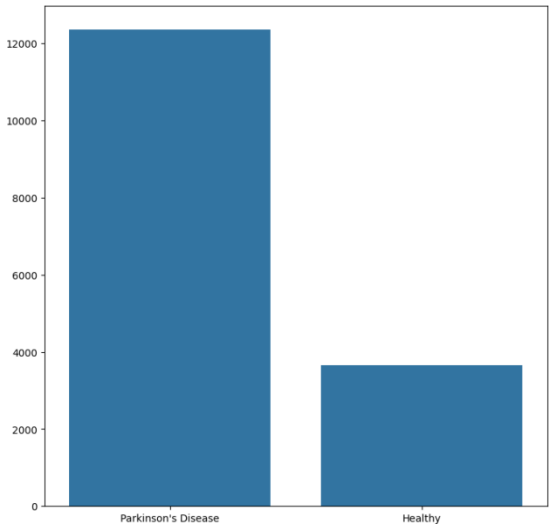

Figure 4. Number of each cohort

Table 2 presents the mean and standard deviation of the ages of individuals diagnosed with Parkinson's Disease compared to healthy individuals. The mean age for both groups is nearly the same, with Parkinson's Disease patients having a mean age of 64.67 years and healthy individuals having a mean age of 64.66 years. However, the standard deviation, which measures the variability or spread of ages around the mean, differs between the two groups. The standard deviation for Parkinson's Disease patients is 9.93 years, indicating a tighter clustering of ages around the mean compared to the healthy group, with a standard deviation of 11.55 years. This difference in standard deviation suggests that the ages of healthy individuals are more widely spread out than those of individuals with Parkinson's Disease.

Table 1. Comparison of Gender Distribution Between Parkinson's Disease and Healthy Individuals

Gender	Female	Male
Parkinson's Disease	7601	4751
Healthy	1383	2269

Table 2. Comparison of Age Distribution Between Parkinson's Disease and Healthy Individuals

Age	mean	standard deviation
Parkinson's Disease	64.67	9.93
Healthy	64.66	11.55

2. Data Preparation

We implemented an intelligent oversampling technique to tackle the class imbalance in our dataset as the first step of data preprocessing. Thereafter, we proceeded with missing value imputation. Missing value imputation is done by Multivariate Imputation by Chained-Equations (MICE). The MICE algorithm is a robust statistical method that addresses missing values by repeatedly estimating values based on the relation- ships among variables. This process reveals hidden patterns within datasets and restores their completeness, enabling through analysis [30]. After missing value imputation, some values were selected randomly to make artificial missing values. According to Snchez-Morales et al. (2017), replacing known values with artificial missing values can improve model performance [31]. Meanwhile, it lets the model get an overview of the dataset and learn model distribution. In the next step, we have generated masks that delineate both artificial and genuine null values. Then, all missing values were imputed using imputation techniques. Additionally, a min-max scaler was applied to aid in better training and convergence of the model.

3. Model Architecture

In this study, we aim to classify input data into two categories Parkinson's disease and healthy while imputing missing values. This dual task requires the model to differentiate between the two groups and to learn the underlying data distribution for effective imputation. We developed a new deep-learning model that can handle imputation and classification tasks in a single end-to-end training framework.

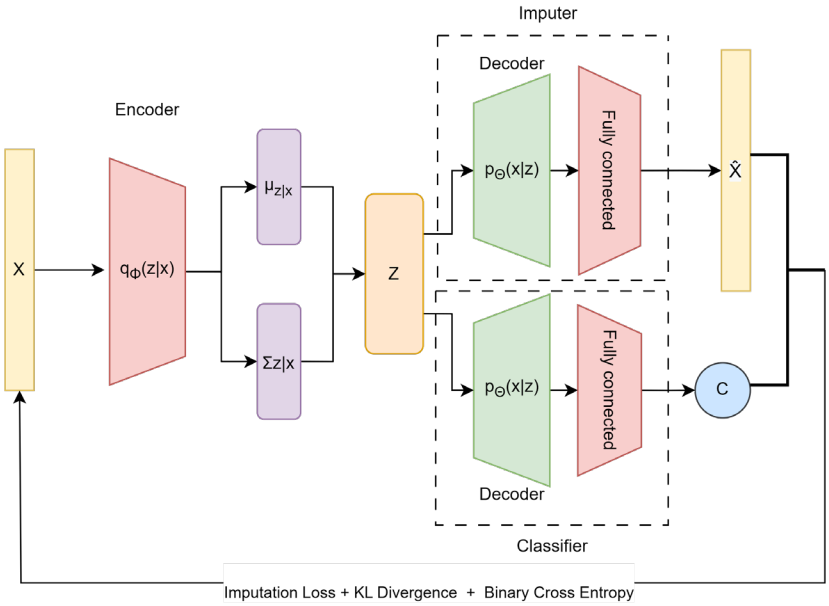

Figure 5. Schema of Fully Connected Variational AutoEncoder

Our newly introduced architecture, termed Variational AutoEncoder-based Imputation and Classification (VICPute), leverages a Variational AutoEncoder (VAE) as its foundational structure [32, 33, 34, 35]. As depicted in 5, the VICPute model incorporates an encoder and two specialized components: an Imputer and a Classifier. The Imputer and Classifier are equipped with their decoders, which share the same architectural framework, and a fully connected layer to fulfill their respective functions. The encoder and decoders are constructed using computational units referred to as Comp Blocks, details of which are further elucidated later in the text. This approach ensures that our model not only classifies data with high accuracy but also effectively manages missing data through imputation, thereby enhancing the overall reliability and performance of the system.

Figure 5 represents the schema of VICPute, designed for handling missing values and classification. The input image $\mathbf{X}$ undergoes encoding through $q_\varphi(\mathbf{z} \mid \mathbf{x})$, where the encoder learns to produce a distribution over the latent space characterized by its mean $\mu_{\mathbf{z}|\mathbf{x}}$ and standard deviation $\Sigma_{\mathbf{z}|\mathbf{x}}$. These parameters are used to sample the latent variable $\mathbf{z}$. The latent variable then passes through two decoders $p_\theta(\mathbf{x} \mid \mathbf{z})$, which attempts to reconstruct the input size $\hat{\mathbf{X}}$ through a series of fully connected layers. Additionally, the latent variable $\mathbf{z}$ interacts with another decoder to generate the number of classes $\mathbf{C}$. The overall loss function combines imputation loss, KL divergence, and binary cross-entropy, ensuring the model imputes missing data, maintains a regularized latent space, and performs classification. According to 3.3.5, the loss function is defined to consider the errors of both classification and imputation. The result would be a model robust to missing values and able to classify each sample. In the following sections, we go through the details of the model.

3.1. Computational Block

Inspired by the ResNet model [29], our computational block (CompBlock(d, h)) uses skip connections. The architecture of a CompBlock [36] with input size d and hidden size h is shown in Figure 6. Consider the input tensor with shape (b, d), where b is the batch size and d is the input dimension. It will go through a dropout layer to prevent overfitting. Then, the result will be given to a linear layer with a Relu activation function to extract features. This layer transforms the input to the hidden dimension h, where h is the hidden size of the CompBlock. The result will be concatenated with the input. This skip connection ensures that even if the feature extractor part of the CompBlock cannot learn richer information than the input, the input information will still be available for the next layers.

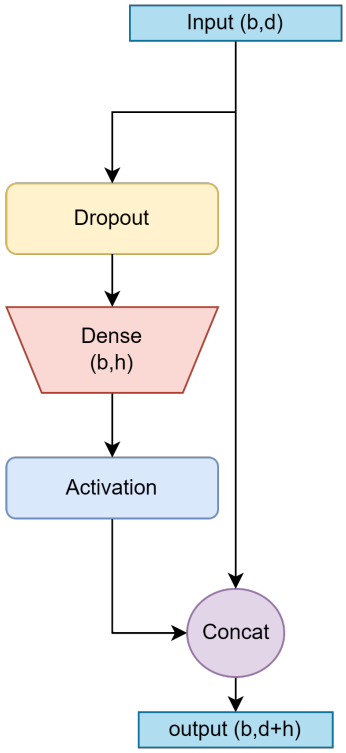

Figure 6. Architecture of Computational Block CompBlock

3.2. Encoder

The encoder plays an essential role in the architecture of VAE models, as it is responsible for calculating the latent space representation used by the decoders in the Imputer and Classifier components of the model. This process directly impacts the performance and the quality of the generated outputs. The encoder in our specified VAE comprises seven stacked CompBlocks, each with the same hidden size.

After the data has passed through the final CompBlock, it reaches two separate dense layers. These layers are designed to calculate the mean (μ) and the logarithm of the variance ($\log(\sigma^2)$) of the latent space distribution. The mean and log variance serve as integral components within the variational segment of the model. Figure 7 represents the parameters of the Gaussian distribution that approximates the actual data distribution in the latent space.

Calculation the mean and log variance is integral to the model's ability to learn a smooth and continuous latent space that

encourages the generation of new, coherent data samples. By training the model to output a mean and log variance, the VAE ensures that the latent variables follow a distribution close to a standard normal distribution ($N(0, I)$). Closing to a standard normal distribution is achieved through the Kullback Leibler (KL) divergence term in the loss function, which regularizes the learned distribution to be as close as possible to the prior (a standard Gaussian distribution).

Once the mean and log variance are obtained, the reparameterization trick allows for backpropagation through the stochastic layer. The reparameterization trick involves sampling from a standard normal distribution and then shifting and scaling this sample using the learned mean and variance. Mathematically, this can be expressed as:

$$z = \mu + \sigma \, 0 \, E, \tag{4}$$

where σ is the exponential of the log variance, and E is a sample from the standard normal distribution. This operation ensures that the sampling process is differentiable and allows gradients to flow through the network, enabling effective model training.

By parameterizing the latent space in this way, the VAE can produce a variety of meaningful latent representations that are input into the decoder and classifier. This parametrization enables the model to reconstruct the input data accurately and to classify the data effectively. The encoder's design and the variational component's precise handling of the mean and log variance are thus fundamental to the overall performance and robustness of the VAE.

3.3. Imputer

The imputer component of our model employs a decoder to impute missing values by effectively recon- structing the input data from its latent space representation, as depicted in Figure 7. This process enables the restoration of incomplete data, thereby facilitating more accurate analysis and classification within the model framework. Like the encoder, the decoder employs seven CompBlocks with the same hidden size, ensuring consistency in architecture and enabling the model to capture and utilize complex patterns in the data. Following the final CompBlock, the Imputer employs a single dense layer to reconstruct the input matrix. This layer is designed to map the latent representation back to the original input size, ensuring that the output is a faithful reconstruction of the original data, completed with imputed missing values. The MLP achieves this by applying a series of learned transformations that decode the compact, abstract features encoded in the latent space into detailed, high-resolution data. The end goal is to produce an output matrix that matches the dimensions of the input matrix, thus providing a complete dataset where missing values have been imputed.

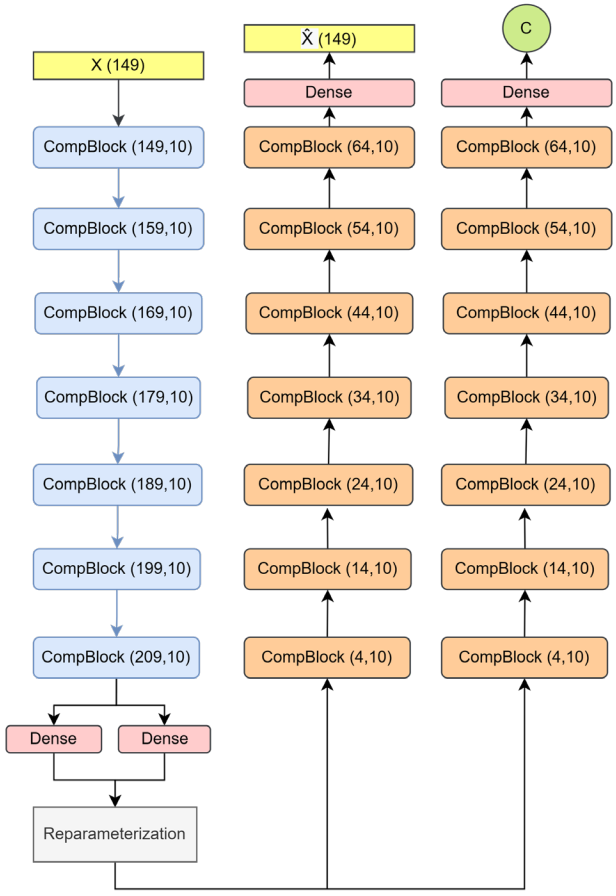

Figure 7. Main model architecture

3.4. Classifier

As shown in Figure 7, the classifier component employs the same computational blocks as the imputer. It comprises seven CompBlocks with the exact hidden sizes, each contributing to the hierarchical representations of features. Given the nature of the task, which entails binary classification, the classifier's architecture is finalized with a single neuron with a sigmoid activation function suited for transforming the output into probabilities. The other branch of the decoder reconstructs the input size tensor which contains implemented values for missing data. Also, the result of encoder is devided Expanding on the design rationale, the choice of CompBlocks, characterized by their ability to extract and abstract features at various levels of granularity, reflects a deliberate effort to capture the nuanced information in the data.

3.5. Loss Function

As described in previous parts, the goals are to impute missing values, classify the dataset, and close the distribution of latent space to normal distribution. Thus, a combined loss function is defined in equation5.

$$Loss = L_{Imputation} + L_{KLD} + L_{BCE}. \tag{5}$$

The equation 5 contains three parts for each goal. In the following, we go into detail about each of them.

To compute the imputation loss, based on [36], the $L_{Imputation}$ is defined as follows:

$$L_{Imputation} = \frac{1}{K+U}\sum_{i=1}^{N}\left(\sum_{j=1}^{M}\left(\hat{x}_j^{(i)} - x_j^{(i)}\right)^2 * \left(1 - M_{miss,j}^{(i)}\right) + 4 * \sum_{j=1}^{M}\left(\hat{x}_j^{(i)} - x_j^{(i)}\right)^2 * M_{drop,j}^{(i)}\right), \tag{6}$$

where $x^{(i)}$ is the value of i-th feature of the j-th data point in the initial dataset (before preprocessing and adding artificial null values), and $\hat{x}^{(i)}$ is the value of i-th feature of model output corresponding to that data point. M_{miss} is a binary mask that identifies all real and artificial missing values, and M_{drop} is a binary mask to determine artificial missing values, which have been dropped randomly in the preprocessing. Moreover, K and U are the total numbers of known and Unknown values, respectively. The first term in equation 6 calculates the reconstruction error on Known values. The second term measures the model error in imputing the artificial missing values by comparing them to their initial true value. According to equation 6, we consider two concepts to train the model. First, we apply the squared error loss function to the known parts of the dataset. Moreover, we train the model to learn the elements we artificially missed. Learning from these dropped values enables the model to handle actual missing values and increases the importance of the dropped values relative to the known values. Hence, we increase their effect by a factor of four compared to the known values. Finally, all values are normalized by $K + U$.

The term L_{KLD} in equation 5 refers to KullbackLeibler divergence (KLD) [33], which is computed by:

$$L_{KLD} = D_{KL}(q(\phi|\theta)\|p(\phi)) = \int q(\phi|\theta)\log\frac{q(\phi|\theta)}{p(\phi)}\,d\phi. \tag{7}$$

KLD is used to find how different the latent space distribution to the normal distribution. In Variational Autoencoder (VAE) models, the KullbackLeibler divergence (KLD) measures how much the learned latent variables approximate a desired prior distribution, typically a standard normal distribution. The KLD term in the VAE objective function acts as a regularizer, penalizing deviations of the learned posterior distribution $q(\varphi|\theta)$ from the prior distribution $p(\varphi)$. Moreover, the KLD term encourages the latent representations to follow the prior distribution, which generates new, realistic samples from the learned latent space. The term BCE in the equation 5 is the binary cross-entropy loss, which is defined as:

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\left[\tau_i \log\hat{\tau}_i + (1-\tau_i)\log(1-\hat{\tau}_i)\right]. \tag{8}$$

where τ_i and $\hat{\tau}_i$ are the true label and the predicted class for i-th data point, $P(\tau_i)$ is the computed probability of the i-th data point to be in class τ_i and N is the total number of samples. The term BCE is used to compute the classification error.

According to the equation 5, the summation of the terms explained in the equations 6 to 8 is computed as the overall loss to cover all aims and let the model learn deeply.

3.6. Weight Initialization

Due to the model's learning enhancement and effects of randomly picking starting weights reduction, the different parts of the model were initialized specially. The encoder's dense layer weights in CompsBlocks were initialized using a principal component analysis (PCA) method that captures most of the important information from the input data. This initialization guarantees that the model performs at least as well as a simpler version that uses PCA. The decoder's and classifier's dense layer weights in

CompsBlocks were initialized using linear regression to enhance their ability to recreate the desired output based on the input they receive. This initialization was done using the entire training dataset at once without creating any additional data variations.

3.7. Metrics

In this part, we demonstrate the metrics for evaluating imputation and classification. We used the f1-score for the classification part.

The F1-score is calculated as the harmonic mean of precision and recall 9, effectively balancing the trade- off between the two metrics. The harmonic average emphasizes the lower values more significantly. If one metric is considerably lower than the other, the harmonic mean will reflect this imbalance more sharply than the simple average. The harmonic mean makes the F1-score particularly useful in situations where false positives and false negatives carry a critical cost, ensuring that the classifier performs well across both performance dimensions. The harmonic mean's sensitivity to the lower value prevents misleadingly high f1-scores when one metric is disproportionately higher, providing a more accurate reflection of the model's overall effectiveness.

$$\text{F1 Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{9}$$

Precision is a metric used to evaluate the accuracy of a classifier in terms of its ability to identify positive instances correctly. It is defined as the ratio of true positive (TP) predictions to the total number of positive predictions made (true positives plus false positives, TP + FP) 10. Precision measures how many of all the instances that the classifier labeled as positive are actually positive. High precision indicates that the classifier has a low rate of false positives, meaning it is reliable when it predicts a positive outcome.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{10}$$

Recall, also known as sensitivity or true positive rate, measures the classifier's ability to identify all relevant positive instances in the dataset. It is calculated as the ratio of true positives (TP) to the total number of actual positive instances (true positives plus false negatives, TP + FN) 11. Recall measures how many instances are actually positive, and how many the classifier correctly identifies. High recall indicates that the classifier successfully captures most of the positive instances, minimizing the number of false negatives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{11}$$

Experiments

In the experimental framework of our study, we partitioned the dataset into two main portions: 85% was used for training and the remaining 15% served as the test set. The training subset was further segregated into training and validation sets at a ratio of 80:20. To facilitate the loss function operations, masks corresponding to the training and validation sets were meticulously generated. To simulate conditions of data scarcity, 30% of the known values within the dataset were randomly omitted to create artificial missing values, for which specific masks were also crafted. The Multivariate Imputation by Chained Equations (MICE) algorithm was employed across all data subsets to address these artificially induced missing values. Additionally, data normalization was achieved using a Min-Max scaler applied across all subsets. The computational architecture was standardized using a hidden size of 10 for all CompBlocks, and the optimization of the model was conducted using the Adam optimizer with a learning rate set at 0.01, complemented by a cosine annealing scheduler to enhance the training dynamics.

Results and Discussion

Effectiveness of the training phase is comprehensively illustrated in Figure 8, which tracks the model's performance, and Figure 9, which shows the progression of binary cross-entropy loss for both the training and validation sets over various epochs. According to Figure 8, the model's performance aligns closely with the binary cross-entropy loss trajectory. Initially, the training loss started at above 1.3 and declined to below 0.6, while the validation loss began just under 0.6 and eventually matched the training loss by the end of the training period. The observed differences between the model performance and the binary cross-entropy loss can be attributed to imputation loss.

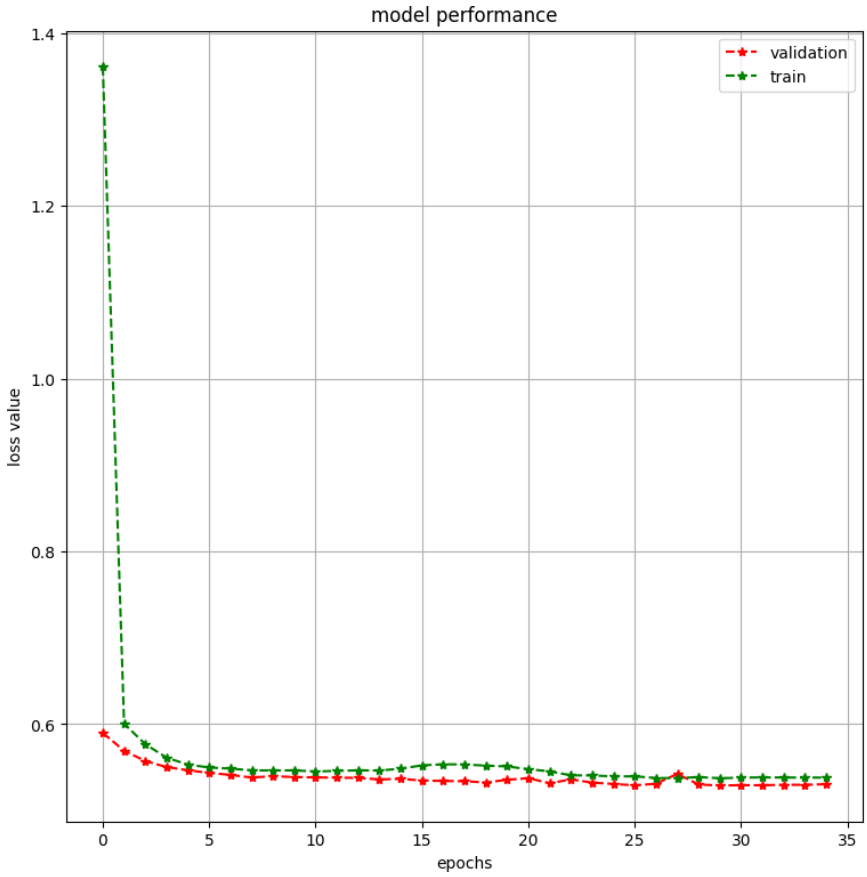

Figure 8. Model Performance

Further depicted in Figure 9, the training dynamics for binary cross-entropy mirrored the overall model performance, completing within 18 epochs. Both training and validation losses experienced a substantial reduction initially before stabilizing. Specifically, training loss commenced at slightly over 0.706 and decreased to below 0.69, whereas validation loss started around 0.7 and converged with the training loss by the final epoch, indicating a consistent reduction across the training process.

Performance metrics revealed an impressive F1-score of 92% for both the training and validation phases, with an F1-score of 87% on the test set, highlighting the model's robustness and effectiveness in processing complex datasets. The subsequent part of our analysis involves comparing our model's performance across different configurations, employing various scalers and imputation methods such as MICE and the KNN Imputer. The comparative analysis is detailed in Table 3.

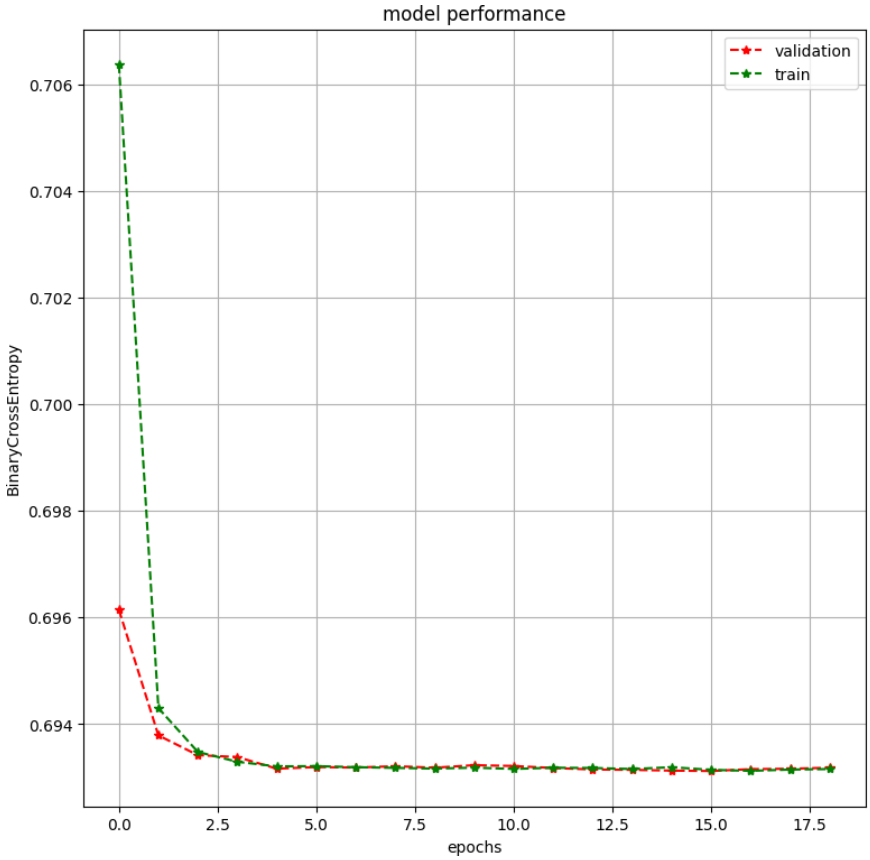

Figure 9. Binary cross entropy for classification part

Table 3 provides a classification report that compares the F1 scores across different imputation techniques combined with various scaling methods. It assesses the performance of both the KNN Imputer and the MICE algorithm under Min-Max scaling and Z-score scaling. The KNN Imputer registered an F1-score of 0.85 with Min-Max scaling and slightly improved to 0.86 with Z-score scaling. Similarly, the MICE algorithm achieved an F1-score of 0.85 with Min-Max scaling and marginally enhanced to 0.87 with Z-score scaling. These findings suggest that both imputation methods yield comparable results with Min-Max scaling; however, the MICE algorithm slightly surpasses the KNN Imputer with Z-score scaling, indicating a potentially more robust performance by MICE in managing missing data when combined with Z-score scaling.

Table 3. The comparison of the classification ability of the model via both KNN Imputer and MICE imputer. Moreover, shows the performance using min max scaler and z-score scaler

Imputer method	min max scaler	z-score scaler
KNN Imputer	0.85	0.86
MICE	0.85	0.87

The results from various machine learning models employed for Parkinson's disease classification on a clinical dataset are consolidated in Table 4. Our model, employing the Fuzzy Clustering Variational Autoencoder (VICPute) technique, demonstrated an accuracy of 87%. Comparatively, Asmae et al. [37], utilizing KNN, achieved a lower accuracy of 79.31%. Berus et al. [38] explored the use of both Artificial Neural Networks (ANN) and Support Vector Machines (SVM) with linear and RBF kernels, reporting accuracies ranging from 85% to 87.5%. Further, the study by Xiong and Lu [25] reported that models based on Logistic

Regression and SVM yielded an accuracy of 88%. In another instance, Masud et al. [39] achieved the highest accuracies in this review, 90% with Logistic Regression and 88% with SVM. Sakar et al. [23] used a combination of binary classifiersachieving accuracies between 84% and 86% across Linear Regression, SVM (Linear and RBF), and KNN. In contrast, Pahuja and Nagabhushan [24] reported varied results with accuracies of 72.81% for KNN, 88.21% for SVM (RBF), and 82.9% for SVM (Linear). Lastly, Mohammed et al. [40] noted accuracies of 86.2% for Logistic Regression and 85.6% for KNN, underscoring the variability and potential of these machine-learning techniques in clinical applications.

Table 4. Comparison of ML Techniques for Parkinson's disease classification on clinical dataset

Study	ML Technique	Accuracy
Our model	VICPute	87%
Asmae et al. [37]	KNN	79.31%
Berus et al. [38]	ANN, SVM (Linear and RBF)	85-87.5%
Xiong and Lu [25]	Logistic Regression, SVM	88%
Masud et al. [39]	Logistic Regression, SVM	88-90%
Sakar et al. [23]	Linear Regression, SVM (Linear and RBF), KNN	84-86%
Pahuja and Nagabhushan [24]	KNN, SVM (Linear and RBF)	72.81-88.21%
Mohammed et al. [40]	Logistic Regression, KNN	85.6-86.2%
Ozturki and Unal [41]	SVM, KNN	73.8-76.8%
Sharanyaa et al. [42]	Logistic Regression, KNN	80.03-90.2%

Conclusion

In this study, the overarching goal is twofold: firstly, to robustly impute missing entries in the dataset, thereby restoring its completeness and enhancing its utility for advanced analyses; secondly, to accurately classify participants into diagnostic categories based on the processed data. Our innovative approach harnesses the capabilities of the VICPute, integrating dual decoders with a shared encoder that facilitates both the imputation and classification processes. Leveraging a comprehensive dataset from the Parkinson's Progression Markers Initiative, our model demonstrated robust capability in imputing missing data and classifying individuals as either healthy or with PD. This achievement was made possible through a sophisticated architecture that combines dual decoders with an advanced encoder, thereby improving the model's accuracy in diagnostic classification and resilience in data imputation. Furthermore, our findings emphasize the potential of AI to enhance the utility and quality of clinical datasets, particularly in neurodegenerative diseases. Additionally, our study opens avenues for future research, including integrating diverse patient data types to enrich the analyses and overcome current limitations, such as lower classification performance and high computational demands. Moreover, this foundational study encourages further exploration into the application of deep learning in clinical settings, aiming to improve diagnostic tools and patient care strategies for Parkinson's disease.

Acknowledgments

We extend our heartfelt gratitude to everyone who helped us improve this work with their helpful comments. Additionally, we are immensely thankful for the insightful comments and suggestions provided by the anonymous reviewers, which have greatly enriched the manuscript.

REFERENCES

1. C.-Y. Lin, Y.-S. Tsai, M.-H. Chang, Impact of olfactory function on the trajectory of cognition, motor function, and quality of life in parkinsons disease, Frontiers in Aging Neuroscience 16 (2024) 1329551. 1
2. J. Blesa, G. Foffani, B. Dehay, E. Bezard, J. A. Obeso, Motor and non-motor circuit disturbances in early parkinson disease: which happens first?, Nature Reviews Neuroscience 23 (2) (2022) 115–128. 1
3. K.-Y. Kwon, S. Park, R. O. Kim, E. J. Lee, M. Lee, Associations of cognitive dysfunction with motor and non-motor symptoms in patients with de novo parkinsons disease, Scientific Reports 12 (1) (2022) 11461. 1
4. X. Deng, Y. Ning, S. E. Saffari, B. Xiao, C. Niu, S. Y. E. Ng, N. Chia, X. Choi, D. L. Heng, Y. J. Tan, et al., Identifying clinical features and blood biomarkers associated with mild cognitive impairment in parkinson disease using machine learning, European Journal of Neurology 30 (6) (2023) 1658–1666. 1
5. J. Jankovic, Parkinsons disease: clinical features and diagnosis, Journal of neurology, neurosurgery & psychiatry 79 (4) (2008) 368–376. 1
6. A. H. Schapira, K. R. Chaudhuri, P. Jenner, Non-motor features of parkinson disease, Nature Reviews Neuroscience 18 (7) (2017) 435–450. doi:10.1038/nrn.2017.62. 1
7. K. Seppi, M. F. Schocke, An update on conventional and advanced magnetic resonance imaging tech- niques in the differential diagnosis of neurodegenerative parkinsonism, Current opinion in neurology 18 (4) (2005) 370–375. 2
8. R. B. Postuma, D. Berg, M. Stern, W. Poewe, C. W. Olanow, W. Oertel, J. Obeso, K. Marek, I. Litvan, E. Lang, et al., Mds clinical diagnostic criteria for parkinson's disease, Movement disorders 30 (12) (2015) 1591–1601. 2
9. A. Mirelman, T. Herman, M. Brozgol, M. Dorfman, E. Sprecher, A. Schweiger, N. Giladi, J. M. Haus- dorff, Executive function and falls in older adults: new findings from a five-year prospective study link fall risk to cognition, PloS one 7 (6) (2012) e40297. 2
10. S. Heinzel, D. Berg, T. Gasser, H. Chen, C. Yao, R. B. Postuma, M. T. F. on the Definition of Parkinson's Disease, Update of the mds research criteria for prodromal parkinson's disease, Movement Disorders 34 (10) (2019) 1464–1470. 2
11. S. Rosenblum, S. Meyer, A. Richardson, S. Hassin-Baer, Early identification of subjective cognitive functional decline among patients with parkinsons disease: a longitudinal pilot study, Scientific Reports 12 (1) (2022) 22242. 2
12. G. Maggi, C. Giacobbe, F. Iannotta, G. Santangelo, C. Vitale, Prevalence and clinical aspects of ob- structive sleep apnea in parkinson disease: A meta-analysis, European Journal of Neurology 31 (2) (2024) e16109. 2
13. D. Belvisi, R. Pellicciari, A. Fabbrini, M. Costanzo, G. Ressa, S. Pietracupa, M. De Lucia, N. Modugno, F. Magrinelli, C. Dallocchio, et al., Relationship between risk and protective factors and clinical features of parkinson's disease, Parkinsonism & Related Disorders 98 (2022) 80–85. 2
14. S. A. Mostafa, A. Mustapha, M. A. Mohammed, R. I. Hamed, N. Arunkumar, M. K. Abd Ghani, M. M. Jaber, S. H. Khaleefah, Examining multiple feature evaluation and classification methods for improving the diagnosis of parkinsons disease, Cognitive Systems Research 54 (2019) 90–99. 2
15. S. A. Mostafa, A. Mustapha, S. H. Khaleefah, M. S. Ahmad, M. A. Mohammed, Evaluating the perfor- mance of three classification methods in diagnosis of parkinsons disease, in: Recent Advances on Soft Computing and Data Mining:

Proceedings of the Third International Conference on Soft Computing and Data Mining (SCDM 2018), Johor, Malaysia, February 06-07, 2018, Springer, 2018, pp. 43–52. 2

16. S. J. Priya, A. J. Rani, M. Subathra, M. A. Mohammed, R. Damaˇseviˇcius, N. Ubendran, Local pattern transformation based feature extraction for recognition of parkinsons disease based on gait signals, Diagnostics 11 (8) (2021) 1395. 2
17. A. M. Ibrahim, M. A. Mohammed, A comprehensive review on advancements in artificial intelligence approaches and future perspectives for early diagnosis of parkinson's disease, International Journal of Mathematics, Statistics, and Computer Science 2 (2024) 173–182. 2
18. K. Marek, D. Jennings, S. Lasch, A. Siderowf, C. Tanner, T. Simuni, C. Coffey, K. Kieburtz, E. Flagg, S. Chowdhury, et al., The parkinson progression marker initiative (ppmi), Progress in neurobiology 95 (4) (2011) 629–635. 2
19. K. Marek, S. Chowdhury, A. Siderowf, S. Lasch, C. S. Coffey, C. Caspell-Garcia, T. Simuni, D. Jen- nings, C. M. Tanner, J. Q. Trojanowski, et al., The parkinson's progression markers initiative (ppmi)– establishing a pd biomarker cohort, Annals of clinical and translational neurology 5 (12) (2018) 1460– 1477. 2
20. K. R. Chaudhuri, P. Martinez-Martin, A. H. Schapira, F. Stocchi, K. Sethi, P. Odin, R. G. Brown, W. Koller, P. Barone, G. MacPhee, et al., International multicenter pilot study of the first compre- hensive self-completed nonmotor symptoms questionnaire for parkinson's disease: the nmsquest study, Movement disorders: official journal of the Movement Disorder Society 21 (7) (2006) 916–923. 2
21. A. H. Karami, S. Rezaee, E. Mirzabeigi, K. Parand, Comparison of pre-training and classification models for early detection of alzheimers disease using magnetic resonance imaging, in: 8th International Conference on Combinatorics Cryptography, Computer Science and Computation, 2023. 2
22. J.-E. Ding, C.-C. Hsu, F. Liu, Parkinsons disease classification using contrastive graph cross-view learn- ing with multimodal fusion of spect images and clinical features, in: 2024 IEEE International Symposium on Biomedical Imaging (ISBI), IEEE, 2024, pp. 1–5. 2
23. C. O. Sakar, G. Serbes, A. Gunduz, H. C. Tunc, H. Nizam, B. E. Sakar, M. Tutuncu, T. Aydin, M. E. Isenkul, H. Apaydin, A comparative analysis of speech signal processing algorithms for parkinsons disease classification and the use of the tunable q-factor wavelet transform, Applied Soft Computing 74 (2019) 255–263. 2, 17
24. G. Pahuja, T. Nagabhushan, A comparative study of existing machine learning approaches for parkin- son's disease detection, IETE Journal of Research 67 (1) (2021) 4–14. 2, 17
25. Y. Xiong, Y. Lu, Deep feature extraction from the vocal vectors using sparse autoencoders for parkin- son's classification, IEEE Access 8 (2020) 27821–27830. doi:10.1109/ACCESS.2020.2973756. 2, 17
26. R. Wang, T. Lian, M. He, P. Guo, S. Yu, L. Zuo, Y. Hu, W. Zhang, Clinical features and neurobio- chemical mechanisms of olfactory dysfunction in patients with parkinson disease, Journal of Neurology 271 (4) (2024) 1959–1972. 2
27. S. H. Lee, S.-M. Park, S. S. Yeo, O. Kwon, M.-K. Lee, H. Yoo, E. K. Ahn, J. Y. Jang, J.-H. Jang, Parkinsons disease subtyping using clinical features and biomarkers: literature review and preliminary study of subtype clustering, Diagnostics 12 (1) (2022) 112. 2
28. F. Murtagh, Multilayer perceptrons for classification and regression, Neurocomputing 2 (5-6) (1991) 183–197. 3
29. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778. 3, 9
30. M. J. Azur, E. A. Stuart, C. Frangakis, P. J. Leaf, Multiple imputation by chained equations: what is it and how does it work?, International journal of methods in psychiatric research 20 (1) (2011) 40–49. 7
31. A. S´anchez-Morales, J.-L. Sancho-G´omez, A. R. Figueiras-Vidal, Values deletion to improve deep im- putation

processes, in: Biomedical Applications Based on Natural and Artificial Computing: Interna- tional Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2017, Corunna, Spain, June 19-23, 2017, Proceedings, Part II, Springer, 2017, pp. 240–246. 7

32. V. Borisov, K. Broelemann, E. Kasneci, G. Kasneci, Deeptlf: robust deep neural networks for het- erogeneous tabular data, International Journal of Data Science and Analytics 16 (1) (2023) 85–100. 8
33. D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013). 8, 12
34. I. O. Lopes, D. Zou, I. H. Abdulqadder, F. A. Ruambo, B. Yuan, H. Jin, Effective network intrusion detection via representation learning: A denoising autoencoder approach, Computer Communications 194 (2022) 55–65. 8
35. C. Zhang, Y. Geng, Z. Han, Y. Liu, H. Fu, Q. Hu, Autoencoder in autoencoder networks, IEEE transactions on neural networks and learning systems 35 (2) (2022) 2263–2275. 8
36. M. Peralta, P. Jannin, C. Haegelen, J. S. Baxter, Data imputation and compression for parkinson's disease clinical questionnaires, Artificial Intelligence in Medicine 114 (2021) 102051. 9, 12
37. F. Cordella, A. Paffi, A. Pallotti, Classification-based screening of parkinsons disease patients through voice signal, in: 2021 IEEE International Symposium on Medical Measurements and Applications (MeMeA), IEEE, 2021, pp. 1–6. 16, 17
38. L. Berus, S. Klancnik, M. Brezocnik, M. Ficko, Classifying parkinsons disease based on acoustic measures using artificial neural networks, Sensors 19 (1) (2018) 16. 16, 17
39. M. Masud, P. Singh, G. S. Gaba, A. Kaur, R. Alroobaea, M. Alrashoud, S. A. Alqahtani, Crowd: crow search and deep learning based feature extractor for classification of parkinsons disease, ACM Transactions on Internet Technology (TOIT) 21 (3) (2021) 1–18. 17
40. M. A. Mohammed, M. Elhoseny, K. H. Abdulkareem, S. A. Mostafa, M. S. Maashi, A multi-agent feature selection and hybrid classification model for parkinson's disease diagnosis, ACM Transactions on Multimidia Computing Communications and Applications 17 (2s) (2021) 1–22. 17
41. S. Ozturk, Y. Unal, A two-stage whale optimization method for classification of parkinsons disease voice recordings, International Journal of Intelligent Systems and Applications in Engineering 8 (2) (2020) 84–93. 17
42. S. Sharanyaa, P. N. Renjith, K. Ramesh, Classification of parkinson's disease using speech attributes with parametric and nonparametric machine learning techniques, in: 2020 3rd International conference on intelligent sustainable systems (ICISS), IEEE, 2020, pp. 437–442. 17