Machine Learning Driven Classification of Neurological Disorders Affecting Human Gait: A Comparative Study

Xiang Xu
University of Michigan
Ann Arbor, Michigan, USA
hsiangxu@umich.edu

Yixin Tang
University of Michigan
Ann Arbor, Michigan, USA
dengzhi@umich.edu

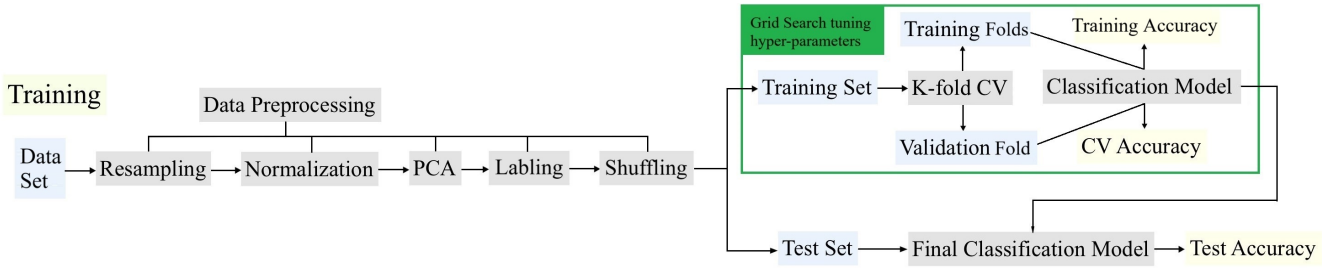


Figure 1: Training and Testing Framework of Gait Classifications Models.

1 Data Preprocessing

The overall data preprocessing section is shown in the left part of Figure 1.

The dataset includes samples from four classes: 57 for Cerebellar Ataxia (CA), 52 for Hereditary Spastic Paraparesis (HSP), 64 for Parkinson's disease (PD), and 65 for healthy class (HC), creating an class imbalance problem that could bias model predictions, which may lead to biased model predictions. To address this, a resampling technique was used, duplicating samples from smaller classes to equalize the number of observations. This resulted in an expanded input matrix of size 238×22, with each row representing a subject and each column a gait parameter.

Anthropometric data (gender, age, weight, height) in the datasets was also incorporated and resampled for balance. The combined matrix of gait and anthropometric features was then normalized using L-2 normalization, ensuring all samples contribute equally and preventing those with larger magnitudes from dominating the model.

Principal Component Analysis (PCA) [1] was applied to reduce the dimensionality from 22 to 8 Principal Components while retaining approximately 97% of the variance of the data. This step improves computational efficiency and model performance by focusing on the most informative features. The dimension of the input data is now reduced to (238×8) from (238×22).

Class labels were assigned as 0 for CA, 1 for HSP, 2 for PD, and 3 for HC. After resampling, the label array matched the 238 samples in the input matrix, ensuring consistent labeling.

Finally, the dataset was shuffled to remove any ordering bias, preparing it for training and evaluation. The balanced and standardized dataset is now ready for model development.

Before training and testing, the dataset is divided into training (80%) and test (20%) sets. Models are trained on the training set, and their performance is assessed on the test set. In training, grid search method is employed with tenfold cross-validation (CV) [2] in training process for systematic hyperparameter optimization. Tenfold CV divides the training set into ten subsets, iterating through each as a validation fold. All possible combinations are explored and those models that achieve the highest CV accuracy is selected. The best-performing model is subsequently evaluated on the test set.

2 Methodology

The overall methodology section is shown in the right part of Figure 1.

To classify disease types, four models are utilized: Support Vector Machine (SVM), Artificial Neural Network (ANN), K-Nearest Neighbors (KNN), and Random Forest (RF). The results of these computational experiments are presented in Section 3.

2.1 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a supervised learning model that finds a hyperplane that optimally separates one class from other classes, maximizing the margin between them, with support vectors being the closest data points to this boundary.

The loss function for an SVM model is:

$$L = \frac{1}{N}\sum_{i=1}^{n} \max(0, 1 - \hat{Y}_i(wX_i - b)) + \lambda\|w\|^2.$$

The first term is the hinge loss, minimizing classification errors by enforcing a margin. The second term is the regularization, balancing model complexity via the regularization hyperparameter λ. Here, N is the sample count. $\hat{Y}_i$ is the true label. w is the weight parameter.b is the bias. They are the two parameters to be learned through training. $(wX_i - b)$ is the predicted output. The kernel trick maps data to a higher dimension for better separation when the original data are not separable by linear decision boundary.

Through grid search, kernel coefficient, regularization parameter [3], kernel types [4] and decision function shapes are optimized.

SVM performs well on complex, high-dimensional data, offering clear decision boundaries and strong generalization, making it effective for our disease classification task. That is why it is selected. However, it may struggle with noisy datasets.

2.2 Artificial Neural Network (ANN)

The Artificial Neural Network (ANN), or Multilayer Perceptron (MLP), is a feedforward model with layers of neurons that learn complex (non-linear) patterns through weights and biases in a supervised manner[5].

ANN aims to minimize the error between its predictions (Y_i) and the ground truth ($\hat{Y}_i$), using a cross-entropy loss function:

$$E = -\frac{1}{N}\sum_{i=1}^{n} \hat{Y}_i \log(Y_i),$$

where $\hat{Y}_i$ is the true label, Y_i is the predicted value, and N is the number of samples. This loss function measures prediction error, and optimization (e.g., via Adam) adjusts the network's parameters to reduce it. Hyperparameters like the maximal iteration times, activation functions used between intermediate layers, batch sizes, solver types (lbfgs, sgd, adam [6]), learning rate properties (constant, invscaling, adaptive) and hidden layer sizes are optimized using grid search.

While ANN excels in learning non-linear relationships, it requires sufficient data to avoid overfitting. It was chosen for its ability to capture complex patterns in gait data for disease classification.

2.3 K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is an instance-based algorithm that classifies samples based on the majority class among their closest neighbors in the feature space.

The classifier compares each test sample to its nearest neighbors in the training data to predict its class.

Grid search is applied to explore number of neighbors considered for classification, which controls local decision granularity. Besides, distance measure such as Euclidean or Manhattan is explored as well as weight configurations which determines neighbor influence on classification [7].

KNN is simple and interpretable, making it a valuable baseline for small datasets. That is why it is chosen. However, it is computationally intensive during inference and less effective on high-dimensional data compared to more complex models.

2.4 Random Forest (RF)

Random Forest (RF) is an ensemble model that builds multiple decision trees, combining their predictions to improve generalization and reduce overfitting.

Each tree is trained on a random subset of features of a random subset of the data, and the final class is determined by majority voting.

Grid search is applied to tune number of decision trees, which controls the ensemble's diversity, limited tree depth to prevent overfitting, the criterion for the quality of a split, and whether bootstrap samples are used [8]. These parameters are set to balance accuracy and generalizability.

RF excels in accuracy and robustness, particularly with small datasets, by reducing variance without overfitting. This method is chosen for its high tolerance to overfitting and robustness, making it well-suited for our limited dataset classification task.

3 Current Results

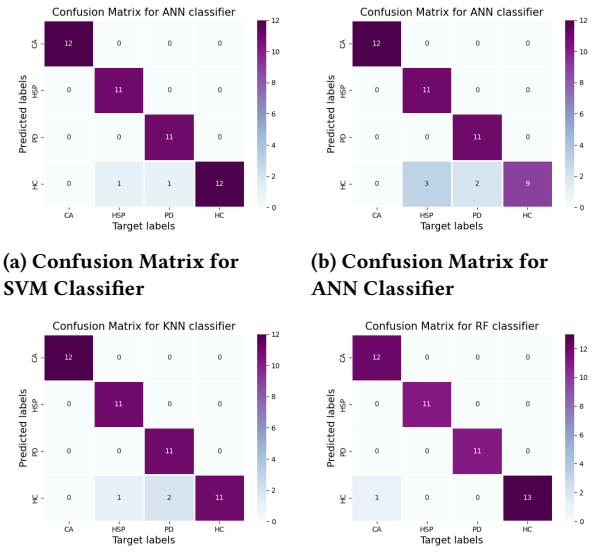
Generally speaking, the accuracy achieved by the algorithms are pretty good even with such a small dataset and such a complex input-output relationship.

Experimental investigations demonstrated that the RF classifier outperformed all other models, achieving highest accuracy. As shown in Table 1, RF got highest CV accuracy at 94.21 % and a 97.92% classification accuracy on the test set. The SVM model also performed well on the test set with an accuracy of 95.83%, followed by ANN and KNN of 93.75% and 89.58%, respectively. Table 1 provides a complete summary of the CV and test accuracies across models.

Table 1: Accuracies for four gait classification models

Method	SVM	ANN	KNN	RF
CV accuracy	91.58	91.58	93.16	94.21
Test accuracy	95.83	93.75	89.58	97.92

Confusion matrices are used to assess detailed classification performance of each algorithm on the test set, which displays counts of predictions across the four gait categories. The confusion matrices for each classifier, shown in Figure 2, reveal distinctive patterns in the misclassification tendencies of each model. Specifically, the SVM, ANN, and KNN classifiers are tend to mislabel cases of HSP



(a) Confusion Matrix for SVM Classifier

(b) Confusion Matrix for ANN Classifier

(c) Confusion Matrix for KNN Classifier

(d) Confusion Matrix for RF Classifier

Figure 2: Heatmaps for four gait classification models

(Hereditary Spastic Paraplegia) and PD (Parkinson's Disease) as HC (Healthy Control), with F_1 scores of 0.92, 0.78, and 0.88, respectively. Random Forest (RF) model primarily mislabels CA (Cerebellar Ataxia) as HC, achieving an overall F_1 score of 0.96.

4 Future Plan for Improvement: Ensemble

In classification tasks, an **ensemble** refers to a group of multiple models whose individual predictions are combined to generate a final output. The main goal of ensemble methods is to boost predictive performance by harnessing the strengths and compensating for the weaknesses of the individual models. According to the results above, the 2 following controllers are planned to be implemented to improve accuracy.

4.1 Soft Voting

Based on the results from the various single classification algorithms analyzed, the Random Forest (RF) exhibits different misclassification patterns compared to the other methods, making soft voting a promising approach to enhance classification accuracy.

Soft voting aggregates the predicted probabilities from multiple classifiers and selects the class with the highest combined probability. Given that the training process of an ANN is considerably longer without substantial accuracy gains, it is more advantageous to combine the RF classifier with either the SVM, the KNN, or both. This combined approach could form a robust meta-classifier utilizing soft voting, potentially improving overall classification performance.

4.2 Bagging

Bagging is an ensemble learning technique that increases model stability and accuracy by training multiple predictors on random subsets of features derived from bootstrapped samples of the original dataset, with outputs typically aggregated via majority voting in classification tasks.

A typical example of bagging is the Random Forest. The superior performance of RF in certain tasks can be attributed to its ability to reduce variance by creating diverse base models and to improve robustness by averaging out the effects of noise and outliers. This mechanism could theoretically help mitigate the noise sensitivity of the SVM and address the reduced performance of KNN on high-dimensional datasets [9]. Therefore, it is worth exploring an alternative classifier where the decision trees in RF are replaced with SVM or KNN models.

References

[1] I. T. Jolliffe, "Principal component analysis," 2005. [Online]. Available: https://api.semanticscholar.org/CorpusID:265962099

[2] T. Fushiki, "Estimation of prediction error by using k-fold cross-validation," *Statistics and Computing*, vol. 21, no. 2, p. 137–146, Apr. 2011. [Online]. Available: https://doi.org/10.1007/s11222-009-9153-8

[3] S. Han, C. Qubo, and H. Meng, "Parameter selection in svm with rbf kernel function," *World Automation Congress 2012*, pp. 1–4, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:3028474

[4] S. Amari and S. Wu, "Improving support vector machine classifiers by modifying kernel functions," *Neural Networks*, vol. 12, no. 6, pp. 783–789, 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608099000325

[5] S. Haykin, "Neural networks expand sp's horizons," *IEEE Signal Processing Magazine*, vol. 13, no. 2, pp. 24–49, 1996.

[6] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:6628106

[7] R. Ghawi and J. Pfeffer, "Efficient hyperparameter tuning with grid search for text categorization using knn approach with bm25 similarity," *Open Computer Science*, vol. 9, no. 1, pp. 160–180, 2019. [Online]. Available: https://doi.org/10.1515/comp-2019-0011

[8] G. SijiGeorgeC and B.Sumathi, "Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction," *International Journal of Advanced Computer Science and Applications*, vol. 11, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:222433720

[9] A. Bertoni, R. Folgieri, and G. Valentini, "Bio-molecular cancer prediction with random subspace ensembles of support vector machines," *Neurocomputing*, vol. 63, pp. 535–539, 2005, new Aspects in Neurocomputing: 11th European Symposium on Artificial Neural Networks. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231204003637