OXFORD JOURNAL
OF STUDENT
SCHOLARSHIP

Precision Medicine Towards Women's Multiple Sclerosis Progression Using Machine Learning Models

Yutong Chen
toni.chen601@gmail.com

ABSTRACT

Over 2.8 million people worldwide are affected by multiple sclerosis, an autoimmune disease in which the immune system attacks the central nervous system. Most individuals with MS experience some degree of neurological disability, and women have an approximately fourfold risk of developing MS compared to men. Diagnosing MS and identifying its subtypes remain challenging and time-consuming, due to varying clinical evaluations across patients. This study investigates whether ensemble optimization improves multi-class MS subtype classification accuracy compared to individual ML models, addressing the research gap in existing approaches that lack ensemble optimization. We preprocessed a publicly accessible, fully anonymized dataset consisting of 273 patient health records to ensure completeness and balance across classes. ML algorithms representing tree-based, neural, and instance-based learning paradigms were developed for multi-class classification in disability score; prevented from overfitting using dropout and regularization; and evaluated using precision, recall, and F1-score. To combine predictions, we implemented ensemble averaging using performance-based weighted voting. The results indicate that assigning higher weights to better-performing models resulted in an accuracy improvement of approximately 2% over simple averaging. Among individual models, the random forest achieved the highest classification accuracy of 90% and the most reliable F1-score. While larger datasets are needed to meet the clinical requirement, these findings demonstrated the potential of machine learning to support clinical decision-making in MS diagnosis, which may be particularly impactful given the higher prevalence of MS in women. The study also highlighted the importance of incorporating gender- and lifestyle-related factors into future clinical and computational studies.

INTRODUCTION

Multiple Sclerosis is an autoimmune disease in which one's body attacks one's own organs. The disease has four phenotypes: clinically isolated syndrome (CIS), relapsing-remitting MS (RRMS), primary-progressive MS (PPMS) and secondary progressive MS (SPMS). Multiple sclerosis symptom progression often changes at different rates for each individual. CIS patients may evolve into RRMS, and the majority of RRMS patients transition into SPMS over time. Doctors specialized in multiple sclerosis examine magnetic resonance imaging (MRI) and complementary health data to diagnose one with

multiple sclerosis and determine the subtype of the disease. However, diagnosis of multiple sclerosis requires time and is sometimes inaccurate at distinguishing and identifying the subtypes. Machine learning models can be used in precision medicine towards multiple sclerosis, assisting doctors to diagnose and classify the disease subtypes within less time while maintaining high accuracy.

By using machine learning, multiple sclerosis, which affects more than 2.8 million people globally, is primarily classified according to clinical symptoms rather than on well-defined pathological mechanisms (Eshaghi et al., 2021). Difficulties in this research include the possibility that the model overfits, in addition to a research gap: few ML research focuses on women's MS progression, and there exists a lack of education regarding correlation between obesity and MS progression.

Most research done examines men and women's symptoms together, using both data to train the model, even though female patients have multiple times the risk and the rate of disease progression as male MS patients. Doctors and scholars have a similar neglect toward how fat percentages or body mass index (BMI) influence the progression of this complicated disease. Few doctors provide dietary or weight advice to multiple sclerosis patients (White 2024). Since there is not a publicly accessible dataset including weight or BMI score with relation to multiple sclerosis progression, the correlation between multiple sclerosis progression will be explored in future studies.

To fill the gap among other studies of multiple sclerosis that lack ensemble optimization, this research uses multiple models such as convolutional neural network, decision tree, random forest, multi-layer perceptron (MLP) neural network, K-nearest neighbors and logistic regression. These models were chosen to represent tree-based, neural, and instance-based learning paradigms in the development process. This study investigates whether ensemble optimization improves multi-class MS subtype classification accuracy compared to individual ML models.

In a person with multiple sclerosis, the nervous system attacks. In one's nervous system, the neurons in the brain, spinal cord, and optical nerves have protective layers called myelin. Myelin is damaged because of attacks by one's own nervous system. This leaves the neuron exposed to even more damage and makes it more difficult to transfer messages. As the nervous system attacks multiple neurons in such a way, messages are blocked in multiple areas in the person's body. Since one's body could not perfectly repair the damaged myelin, many lesions, or scars, remain in the nervous system as the disease progresses. Most patients are first diagnosed with multiple sclerosis in ages of 20 to 40. Their symptom progression rates vary, but factors such as vitamin D deficiency, obesity, smoking or ancestry contribute to faster disease progression (Mayo Clinic).

MATERIALS AND METHODS

Datasets

First, we uploaded the dataset, a CSV file from Kaggle to Google Colab, found in the link https://www.kaggle.com/code/desalegngeb/predictors-of-multiple-sclerosis-disease/input.

A) The dataset we use is publicly accessible and fully anonymous health information of MS patients. Below are all of the variables which contribute to multiple sclerosis progression, which in our study is modeled by the final Expanded Disability Status Scale (EDSS). Each individual has reported all of the independent and dependent variables listed below:
    a) Gender (1=male, 2=female). The female sex is associated with higher odds of rapid weight gain. Women with MS are especially susceptible to rapid weight gain, both at diagnosis and later in the disease course (Conway et al., 2024).
    b) Age of patients in each case is shown in integers.
    c) Schooling is the time the patient spent in school in years.
    d) Breastfeeding (1=yes, 2=no, 3=unknown) is considered a protective factor for the symptom progression of women. Women with multiple sclerosis who breastfeed are half as likely to experience relapse of multiple sclerosis.
    e) Initial symptom 1=visual, 2=sensory, 3=motor, 4=other, 5= visual and sensory, 6=visual and motor, 7=visual and others, 8=sensory and motor, 9=sensory and other, 10=motor and other, 11=Visual, sensory and motor, 12=visual, sensory and other, 13=Visual, motor and other, 14=Sensory, motor and other, 15=visual, sensory, motor and other.
    f) Mono- or Poly-symptomatic 1=monosymptomatic, 2=polysymptomatic, 3=unknown. This column determines the number of symptoms.
    g) Oligoclonal_Bands 0=negative, 1=positive, 2=unknown. Proteins that indicate inflammation in the central nervous system, Oligoclonal Bands can be signs of multiple sclerosis.

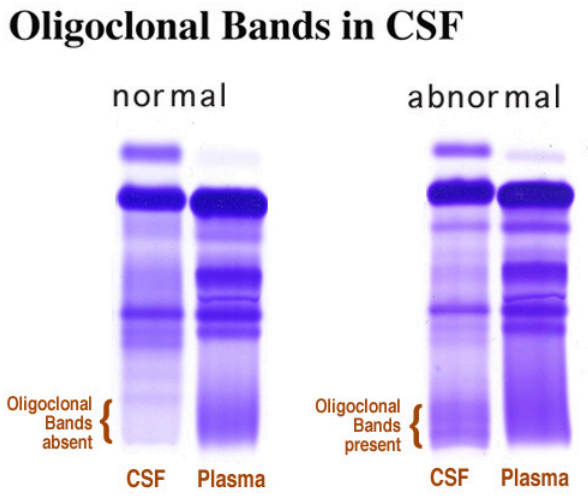

Figure 1: Comparison of normal and abnormal oligoclonal bands in cerebrospinal fluid(CSF) of people with multiple sclerosis. The presence of oligoclonal bands serves as an indicator of multiple sclerosis.

    h) Long-Latency Somatosensory Evoked Potentials (LLSSEP) 0=negative, 1=positive
    i) Upper limb somatosensory evoked potentials (ULSSEP) 0=negative, 1=positive
    j) Visual evoked potential (VEP) 0=negative, 1=positive

k) Brainstem auditory evoked potential (BAEP) 0=negative, 1=positive
l) Periventricular MRI 0=negative, 1=positive
m) Cortical MRI 0=negative, 1=positive
n) Infratentorial MRI 0=negative, 1=positive
o) Spinal Cord MRI 0=negative, 1=positive
p) Initial EDSS is a measure of disability using scores 1 to 10 that the patient is initially discovered to have.
q) Final EDSS is a measure of disability using scores 1 to 10 that the patient is currently discovered to have.
r) Group 1=clinically definite multiple sclerosis(CDMS), 2=non-clinically definite multiple sclerosis (non-CDMS)

The independent variables are the patients' Initial_Symptom, Mono_or_Polysymptomatic, Oligoclonal_Bands, LLSSEP, ULSSEP, VEP, BAEP, Periventricular_MRI, Cortical_MRI, Infratentorial_MRI, Spinal_Cord_MRI, and Initial_EDSS. These independent variables provide biological and demographic indicators of risk in multiple sclerosis. The dependent variable to predict is the Final EDSS score, which models the current multiple sclerosis stage of the patient.

Train-Test Split

Machine learning models will train from a dataframe, learning various inputs and outputs, so that it predicts the correct output when encountering new data. However, if the model could access all of df, the model could memorize the training data instead of learning from it and we would not be able to identify that as we would not have additional data to test the model. We need a train-test split in order to evaluate the models when they encounter datasets they have not seen. In our approach, 67% of all data is used for training and 33% for testing.

Ensemble averaging

To classify the final EDSS score of multiple sclerosis patients, we created 4 different classification models.

Decision Tree

First, we use a decision tree classifier. The branches of the decision tree represent answers to a question at the node. Starting from the root node of the tree, the tree branches out into different possibilities based on features in the data. The decision tree's diverse applications include medical diagnosis or prediction of exam results.

The length of the training set is 74. $\log_2(74) \approx 7$, rounded up to the least integer, so the max_depth value needed is 7 for all data to end up in different leaves of the decision tree. When max_depth = 7, the

training accuracy of different data is theoretically 1. However, the experimental accuracy was not perfect, 0.973. This may be because some data has the same inputs but different outputs.

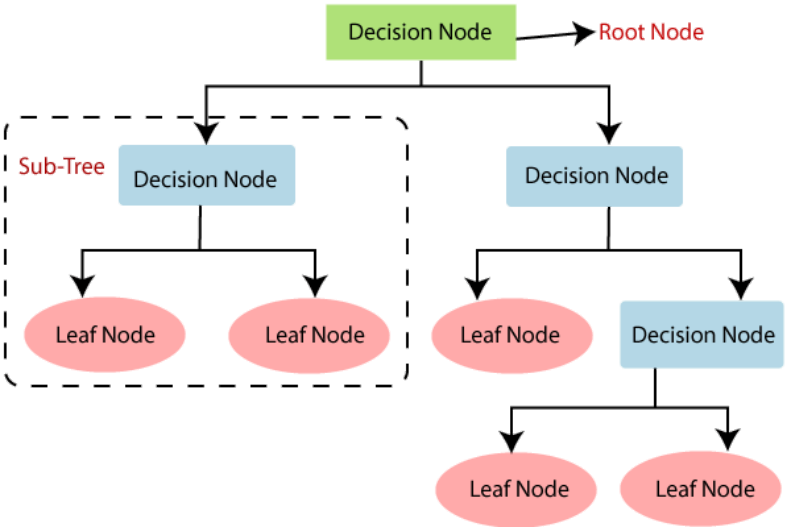

Figure 2: Structure of a decision tree in which based on decisions, the root node extends downward through branches to reach a leaf node (*Decision Tree Learning in Machine Learning*).

Random Forest

Second, we use a random forest classifier, which consists of multiple decision trees used for model training. In addition to the features of a decision tree, random forests gather results from all trees and predict the majority votes. Random forest picks a few columns at random to decide how to split the data. This randomness helps the trees stay different from each other.

A random forest that contains multiple such decision trees is not guaranteed perfect. Each decision tree only trains from a fraction of the whole data and different decision trees have different training data. Therefore, when all decision trees in a random forest are combined, they would give incorrect predictions about data they have not seen.

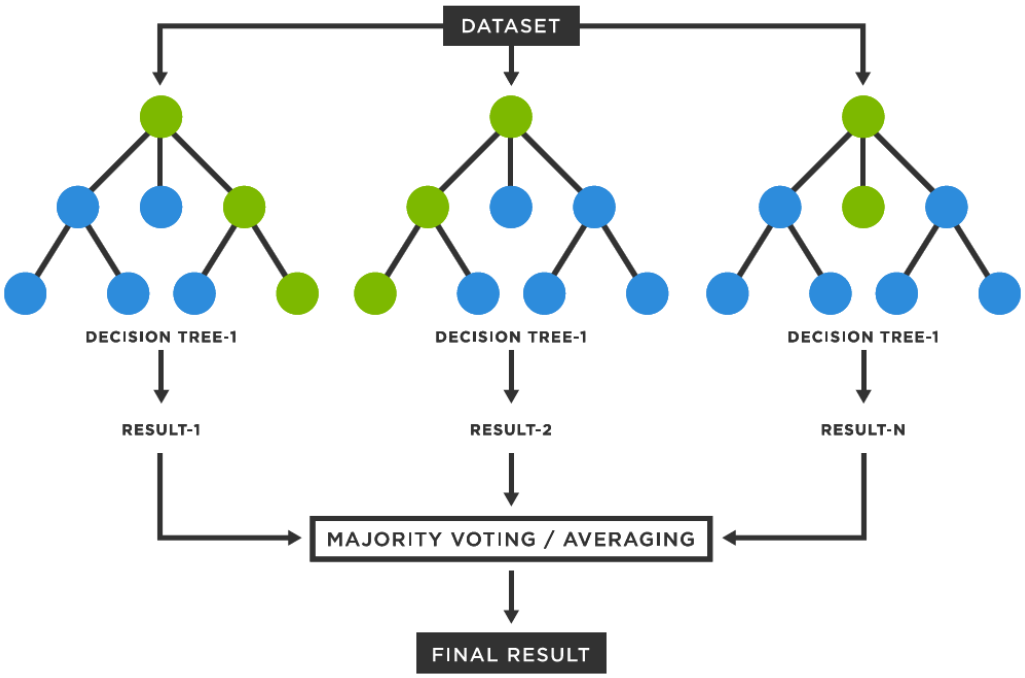

Figure 3: Diagram showing the algorithm of a random forest classifier, taking multiple decision trees into majority voting or averaging to predict the final result (*Miro Medium*).

Multi-Layer Perceptron Classifier(MLP)

Third, we use a Multi-Layer Perceptron(MLP) Classifier. MLP is a neural network for classification imported from ScikitLearn. It initialized weights and biases with small random values. During forward propagation, data is passed through layers of the neural network, including the hidden layers. The activation function introduces non-linearity helping the network learn complex relationships. Then, the network updates its weights and biases to minimize the loss using optimization. The model fits X_train with y_train. It then predicts y values using X_train and using X_test, resulting in different accuracies in training and testing. The training accuracy is about 15% higher than the testing accuracy.

One major limit of AI/ML models used in clinical settings is that they perform inconsistently when data is collected from other medical scanners or machines, since data from different sites look differently. When the model memorizes exactly how data should look, instead of learning the patterns and relationships, overfitting occurs. Therefore, we use the dropout technique to prevent overfitting. Some parts of the data are hidden to the model. If the original model assigns big weights to those parts, removing those parts would result in mistakes. That means the model is overfitting and a new model/approach is needed. The model should predict using a combination of parts to predict, not just focusing on one aspect. The more complex the model, the more likely to overfit, and the dropout should be bigger.

Specifically for the multi-layer perceptron neural network, we used regularization to reduce overfitting.

Some models will overfit even after train-test-split, for instance, assign a huge weight to one certain hyperparameter and leave the rest to zero. To prevent overfitting, the MLP Classifier uses regularization, an alpha value of 0.0001 that penalizes the model for every high weight it assigns. L1 Regularization penalizes the model for the sum of weights it assigns. Therefore, the model would penalize a lot for every high weight it assigns. On the other hand, instead of penalizing the model for the sum of weights it assigns, L2 regularization penalizes by the sum of these weights squared. Compared to L1 regularization, the penalty of L2 regularization punishes overfitting at a larger scale, making L2 regularization more appropriate to prevent overfitting. As a result of L1 or L2 regularization, the model only assigns big weights that are necessary for high accuracy.

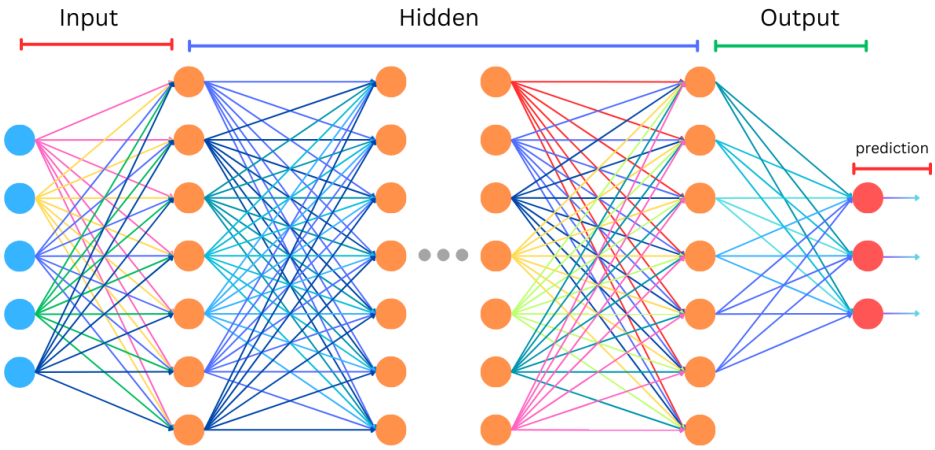

Figure 4: A demonstration of a multi-layer perceptron neural network algorithm with inputs and multiple hidden layers that generates an output out of three predictions (*MLP-Illustration* 2024).

K nearest neighbor(KNN)

Fourth, we use a K nearest neighbor classifier, which compares values of the testing data to the nearest n data around this testing data. The testing data which the model has never seen before is classified into the categories with the closest values. This model fits X_train with y_train. It then predicts y values using X_train and using X_test, resulting in different accuracies in training and testing. The training accuracy is about 15% higher than the testing accuracy.

Our classification models compare the values of the new testing data points with K numbers of completely classified training datasets. For example, if K = 5, the classification model finds the most similar 5 training data points and takes the majority voting of the categories of the 5 points. If 3 or more data points among the five are in category 1, then the new data would be predicted as category 1, as well. After experimenting with multiple values of K, the predictions using each K are taken into majority voting to output the final prediction.

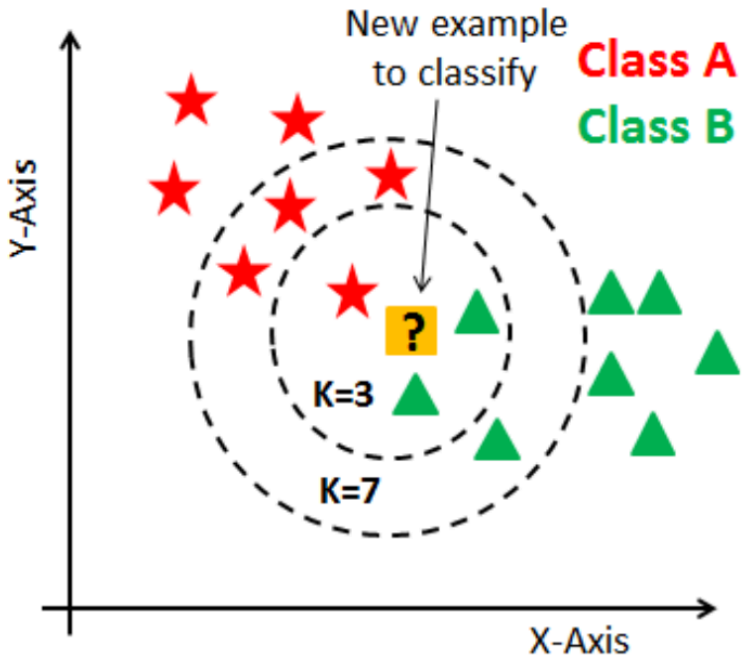

Figure 5: A demonstration of a k-nearest neighbor algorithm that classifies a new value based on the number of closest data points in each category.

We use precision, recall and f-1 score to evaluate each model. Precision is the percentage of accurate predictions among all the predictions that are positive. It can be calculated using the formula:

$$\text{Precision} = \frac{TP}{TP+FP}$$

where TP is the number of true positive predictions and FP is the number of false positive predictions.

Recall is the percentage of accurate predictions among all the actual positive values. It can be calculated using the formula:

$$\text{Recall} = \frac{TP}{TP+FN}$$

where TP is the number of true positive predictions and FN is the number of false negative predictions.

F-1 score is a metric that combines precision and recall. It can be calculated using the formula:

$$\text{F-1 score} = \frac{2(precision \cdot recall)}{precision+recall}$$

Combining the models

We used 3 ways to combine the 4 models' predictions using ensemble averaging. Then, the resulting performances of each combined model are compared and evaluated.

One way to combine is to take the numerical average of predictions. This method assigns equal weights to all four models during testing. The output has accuracy 0.773, which is equal to the testing accuracy of KNN, the model that has the lowest testing accuracy. This overall decrease in accuracy is due to the multi-category nature of this problem. Instead of binary classification, the output is categorized in EDSS score from 1 to 3. For instance, if two models predicted 1 and two models predicted 3, the average prediction would be 2. However, none of the 4 models predicted 2. Alternate approaches are needed.

Assigning weights to the models that are better at predicting results in more effective predictions than taking the average prediction of all models. Each of the 4 models can be assigned weights when their results are combined. Their weights are calculated using the formula:

$$w = accuracy_{test} / accuracy_{test,total}$$

Since each weight of model is a decimal from 0 to 1 and all 4 weights sums to be 1, the combined prediction all_model_pred2 is:

$$y_{pred2,all} = y_{pred,dt} \cdot w_{dt} + y_{pred,rf} \cdot w_{rf} + y_{pred,mlp} \cdot w_{mlp} + y_{pred,knn} \cdot w_{knn}$$

However, the testing accuracy of the model is unknown if the models have no access to the y_test.

Instead, we combined predictions without knowing their testing accuracy beforehand. By default, the weight of each model is equal to 1. For every mistake each model makes, the weight of that model reduces by a fixed factor. This factor can increase or decrease based on the emphasis on mistakes. In this program, we picked the factor of 10/11. In this way, the final weights of each model can be determined after repeating this procedure throughout the dataset X_test. Then, the majority prediction using these weights - whichever value from 1 to 3 has the highest weights - is stored as the combined prediction.

RESULTS

We found that assigning higher weights to better-performing models resulted in an accuracy improvement of approximately 2% over simple averaging. For example, the highest individual model achieved an accuracy of 0.66. Meanwhile, the combined model, created by decreasing the model's weight by a factor of 10/11 for each mistake, achieved an accuracy of 0.68.

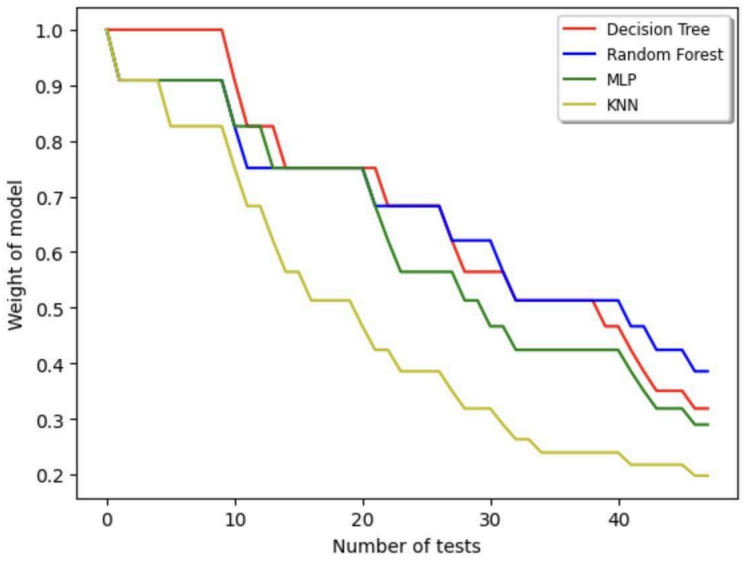

Figure 6: Weight of 4 different models versus the number of testing data. As each model makes a mistake in prediction, its weight decreases by a factor of 10/11.

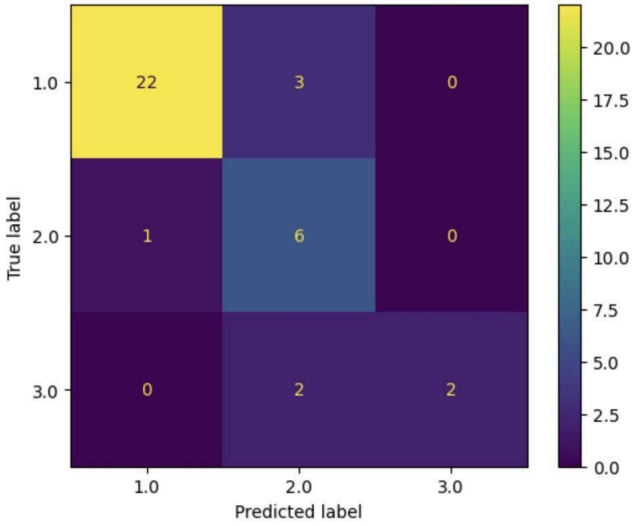

Figure 7: Confusion matrix for decision tree classifier

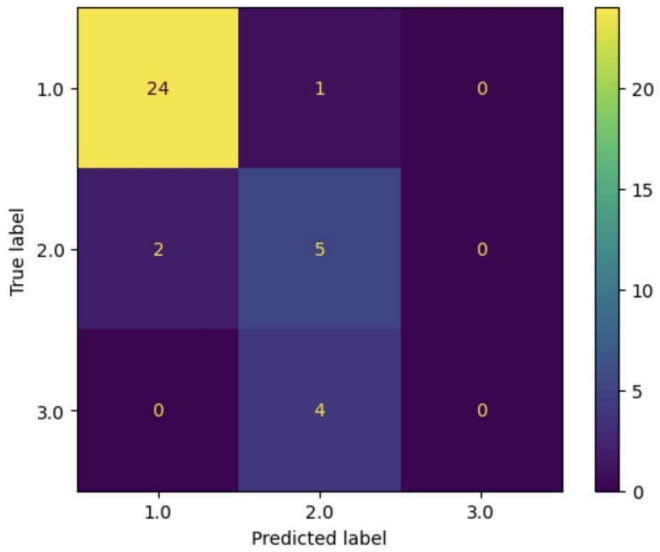

Figure 8: Confusion matrix of random forest classifier

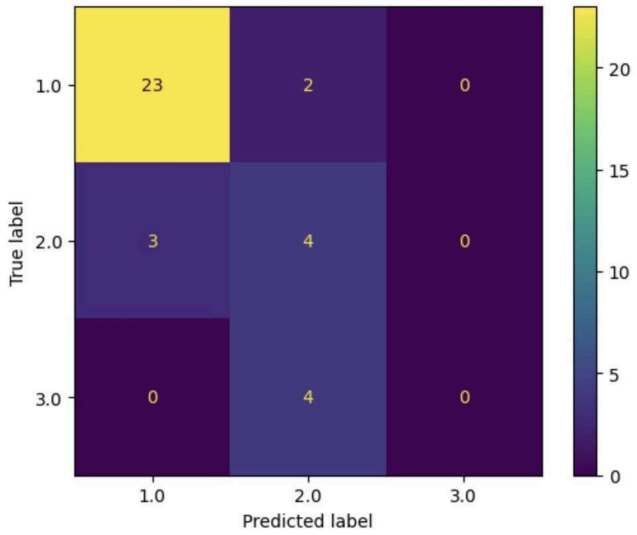

Figure 9: Confusion matrix of multilayer perceptron classifier

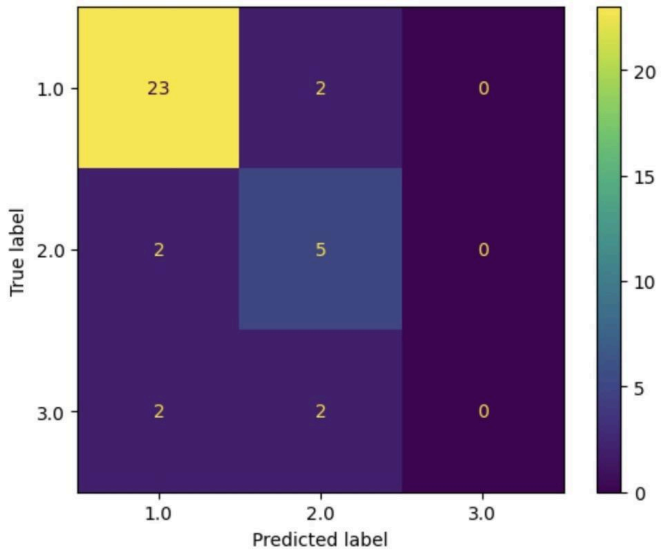

Figure 10: Confusion matrix of k-nearest neighbor classifier

DISCUSSION AND CONCLUSION

Multiple sclerosis is an autoimmune disease that causes the body to attack its own nervous system, leading to a wide range of symptoms and disability levels. Over 2.8 million people globally are diagnosed with multiple sclerosis, and the majority of patients suffer from disability associated with the disease. Moreover, women have about 4 times the risk of getting multiple sclerosis than men. Diagnosing multiple sclerosis and identifying its subtypes—clinically isolated syndrome, relapsing-remitting, primary-progressive, and secondary-progressive—can be time-consuming and difficult, as doctors rely on MRI scans and clinical evaluations that vary between patients. This study explores how machine learning (ML) models can assist in diagnosing and classifying MS subtypes more efficiently and accurately. Using patient health data and MRI scans from a public dataset, several ML algorithms were developed and compared, including decision tree, random forest, multilayer perceptron, k-nearest neighbors, and convolutional neural network (CNN) models. The models were evaluated using precision, recall, and F1-score metrics. To improve overall accuracy, ensemble averaging techniques—such as weighted voting based on models' performances—were used to combine predictions from different models. The results show that assigning higher weights to better-performing models produces more reliable predictions than simple averaging. Overall, this research demonstrates that machine learning can enhance the accuracy and efficiency of MS diagnosis, especially for women, and highlights the importance of incorporating gender- and lifestyle-related factors into future clinical and computational studies. Using 4 machine learning models, the highest accuracy we achieved is 90% using random forest.

Machine learning models in the clinical setting require nearly 100% accuracy. This research achieved the highest accuracy of 90% using a random forest classifier. Still, this accuracy does not satisfy clinical requirements due to limited publicly accessible and anonymous data. Further research in this field can

improve on the models' accuracy, precision, and recall to qualify for implementation in the clinical setting by using larger datasets. Future studies in this field can also add additional parameters such as patients' vitamin D intake, weight, and ancestry.

REFERENCES

AI Chronicles: When Machines Do It Better. (2024). *MLP-Illustration*. Dinocausevic. Retrieved November 8, 2025, from https://dinocausevic.com/wp-content/uploads/2024/05/MLP-Illustration.png.

Algorithm of a K-Nearest Neighbor Classifier. (n.d.-a). Miro Medium. Retrieved November 8, 2025, from https://miro.medium.com/v2/0*ItVKiyx2F3ZU8zV5.

Conway, D. S., Toljan, K., Harris, K. A., Galioto, R. M., Briggs, F. B. S., & Hersh, C. M. (2024, December 12). *Body mass index trends over four years in persons with multiple sclerosis - sciencedirect*. Science Direct. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S2211034824007946

Eshaghi, A., Young, A. L., Wijeratne, P. A., Prados, F., Arnold, D. L., Narayanan, S., Guttmann, C. R. G., Barkhof, F., Alexander, D. C., Thompson, A. J., Chard, D., & Ciccarelli, O. (2021). Identifying multiple sclerosis subtypes using unsupervised machine learning and MRI data. *Nature Communications*, *12*(1). Retrieved from https://doi.org/10.1038/s41467-021-22265-2

Mayo Clinic. "Mayo Clinic Explains Multiple Sclerosis." *YouTube*, YouTube, 5 Jan. 2022, www.youtube.com/watch?v=Z1ibVlGflPs&t=125s.

News-Medical. (2025, April 29). *Researchers discover new biomarker to predict multiple sclerosis progression*. Retrieved from https://www.news-medical.net/news/20250429/Researchers-discover-new-biomarker-to-predict-multiple-sclerosis-progression.aspx

"Predictors of Multiple Sclerosis Disease." *Kaggle*, Kaggle, 21 May 2023, www.kaggle.com/code/desalegngeb/predictors-of-multiple-sclerosis-disease/input.

Rose, J., & Houtchens, M. (n.d.). *Oligoclonal Bands in CSF in a Patient with Multiple Sclerosis*. Eccles Health Sciences Library. The University of Utah. Retrieved November 8, 2025, from https://library.med.utah.edu/kw/ms/mml/ms_oligoclonal.html.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., & Department of Computer Science, University of Toronto. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. In Y. Bengio (Ed.), *Journal of Machine Learning Research* (Vol. 15, pp. 1929–1958). Retrieved from https://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf

Structure of a Random Forest Classifier. (n.d.). Miro Medium. Retrieved November 8, 2025, from https://miro.medium.com/v2/1*R3oJiyaQwyLUyLZL-scDpw.png.

TrainTestSplit. (n.d.). *Decision Tree Learning in Machine Learning*. TrainTestSplit. Retrieved November 8, 2025, from https://traintestsplit.com/wp-content/uploads/decision-tree-learning-in-machine-learning.png.

White, T. (2024, February 28). *Body fat and MS, the evidence is now plain to see*. Overcoming MS. Retrieved from https://overcomingms.org/latest/body-fat-and-ms-evidence-now-plain-see-0#Study-limitations__5