Detecting Parkinson's Disease using Machine Learning Models

Pranathi Alluri
Northeastern University, Boston, MA, USA

Abstract

Parkinson's disease (PD) is the second most common neurodegenerative disease and affects ten million people around the world. With both motor and non-motor symptoms, Parkinson's has a major impact on simple tasks, as well as the daily lives of many. Although there is currently no cure, treatments tend to be effective when patients are diagnosed early on. Thus, creating machine learning models running classification on various physiological measurements of people with and without PD, can help in detecting and predicting Parkinson's disease early-on, hopefully improving the lives of patients. In this paper well-known supervised machine learning models for classification, including logistic regression, ridge classification, support vector machines (SVMs), decision tree, random forest, and feed forward neural network classification, are applied to biomedical voice measurements data of people with and without PD in an effort to choose the most effective hyper parameters for each model, as well as the best model in general. It was found that all models performed well, however SVMs had the highest accuracy overall.

1 Introduction

Parkinson's disease is a common motor system disorder caused by the degeneration of dopaminergic neurons in the substantia nigra, located in the midbrain. Although symptoms can vary between patients, the most common ones include tremor, slowness of movement (bradykinesia), rigidity, and stiffness. However, most of these larger motor symptoms only become evident later on, after 60 to 80% of dopaminergic neurons have already been lost or impaired [6]. As a result, treatments may not be as effective and achieving a great quality of life becomes more difficult for patients with PD. Hence, it is important to be able to detect the early signs of PD and provide a diagnosis early on. However, this is challenging as there currently isn't a specific test for diagnosis.

One of the most common signs of PD early on is a soft or low voice. Ninety percent of PD patients exhibit some form of vocal disorder in the early stages of the disease [4], because of problems with muscle activation in the respiratory system [5] and stiffness of vocal cords. As a result, creating a machine learning model that uses various features of patient voice recordings to detect Parkinson's disease offers a solution to the present concerns in diagnosing Parkinson's disease.

Finding the best binary classification model offers many benefits to both doctors and patients. For one, this increases the options for telediagnosis and telemonitoring systems for PD, reducing inconvenient physical visits to doctor offices for patients and assisting in

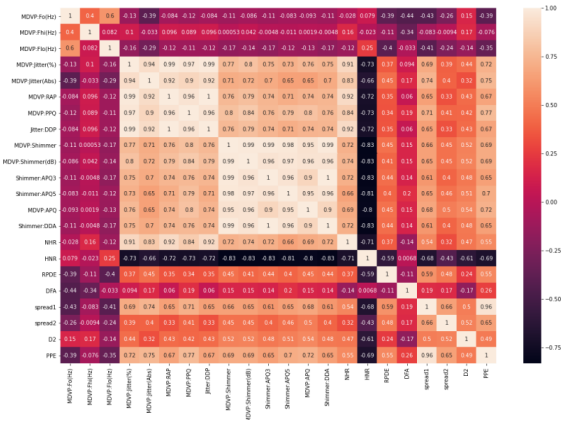
diagnosis, as well as reducing workload for clinicians [4]. Furthermore, voice recordings are easy to obtain, providing noninvasive data which can be used to detect the disease early on, leaving more options for treatments, such as L-dopa and deep brain stimulation. Overall, the aim is to create an accurate model that benefits and improves the lives of Parkinson disease patients.

2 Methodology

To develop the best model for binary classification various common supervised machine learning models, such as logistic regression, SVM, and neural networks, will be run on a pre-collected dataset that contains biomedical voice measurements.

2.1 Data Description

This dataset was created by Max Little of the University of Oxford in collaboration with the National Center for Voice and Speech and is provided by the UCI machine learning repository. This is an extremely popular dataset for the classification of PD, as there are multiple samples collected from 23 patients with Parkinson's. Specifically, there are 22 numerical features characterizing the voice recordings, which will be used to run classification, including: linear and nonlinear measures of vocal fundamental frequency, measures of variation in amplitude, ratio of noise to tonal components, nonlinear dynamical complexity measure, and the signal fractal scaling exponents. The "status" attribute will be used for classification, with 1 indicating a person with PD and 0 indicating a healthy person. There are no null values or need for preprocessing before running the models. While this dataset is well referenced, it is limited due to the small number of subjects, which will be addressed by including a larger Parkinson's disease classification dataset also stored in the UCI repo. [2][3][4]. After running exploratory data analysis, it is observed that the data is normally distributed, as seen in the figure.



2.2 Creating Machine Learning Models

Before running the machine learning models the data is normalized using min-max scaler and split into 80% training and 20% testing. In order to run hyperparameter tuning, a validation set is needed, for which we will be using k-fold cross validation with 10 splits.

2.2.1 Logistic Regression

Logistic regression is a common linear model for classification, which works well when the classes are linearly separable. The main equation used is the sigmoid function, which maps input to a value between 0 and 1. There are several solvers that can be used to run

logistic regression. Since the dataset is relatively small, “newton-cg”, “lbfgs”, and “liblinear” solvers will be looked at.

Newton-cg is a solver that uses a modified Newton’s method to find optimal weights and is the default solver for logistic regression. Similarly, lbfgs uses the L-BFGS algorithm to find the optimal weights. Both the solvers use L2 regularization as the default penalty. On the other hand, liblinear is the default solver for logistic regression with L1 regularization used for the penalty/error term and uses a coordinate descent algorithm to find the weights.

Along with picking an optimal solver, an optimal value needs to be picked for the regularization strength parameter (C), which controls the amount of regularization applied to the model, with smaller values resulting in stronger regularization.

2.2.2 Ridge Regression

Ridge regression is a variant of logistic regression that adds a penalty term to overcome issues with over fitting. While the previous implementation of logistic regression included L2 regularization, making it equivalent to ridge regression, by using a more specific model the ideal parameters for regularization strength can be set easily. In other words, the effect of various alpha values can be measured, with larger alpha values resulting in a greater importance for the regularization term, which in turns reduces the model’s complexity. On the other hand, a lower alpha value will diminish the effect of the L2 regularization term and may cause overfitting.

2.2.3 SVM

Unlike logistic regression which models the probability that a given data point belongs to a given class, SVM finds the hyperplane that best separates the two classes. Specifically, soft margin SVM is used to allow for some misclassification errors in order to better generalize on the test data.

There are 2 hyperparameters that need to be investigated – the regularization parameter (C) and gamma, which controls the width of the kernel. In general, the goal is to maximize the margin, while minimizing classification errors.

2.2.4 Decision Tree

Decision trees recursively partition data into classes by selecting features that best split the data. Various techniques can be used to train the data, including limiting the maximum depth and minimizing the number of samples required to be at a leaf node, which are the specific hyperparameters that will be focused on.

2.2.5 Random Forest

To create an instance of the random forest, there will be a focus on the number of estimators, which refers to the number of decision trees that are trained on different subsets of the training data. Furthermore, the best evaluation for the maximum number of features to be considered when looking at the best split at each node for the decision

trees will be investigated. Specifically, the square root and log2 of the total number of features.

2.2.6: Feed Forward Neural Network

The last model that will be used for classification is a feed forward neural network. To begin, we will split the training data into 60% train and 20% validation. From there the number of hidden layers, number of neurons at each layer, and various epochs will be observed to create the best network. Relu will be used for the hidden layers with a classification for the output layer.

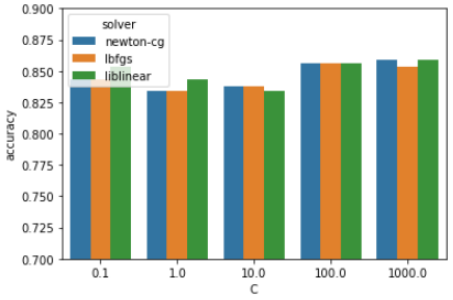
In general, cross validation will be used to investigate the effect of the listed hyperparameters and pick the best one. Once the best parameters are set and the models are trained, classification reports will be run on each model, to compare the various models to pick the best one overall.

3 Analysis

3.1 Evaluation of hyperparameters

3.1.1 Logistic Regression

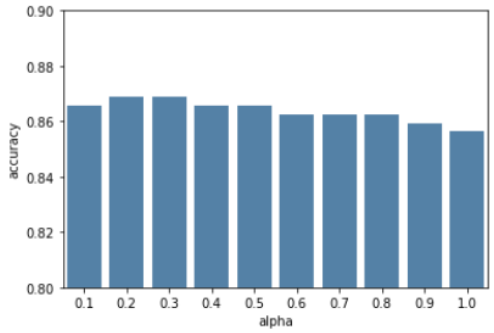
After running cross validation on logistic regression, the best model uses a regularization strength parameter of 1000 and a newton-cg solver. This indicates that a good model doesn't need such a strong weight placed on L2 regularization. This is seen by the high accuracy with C values of 100 and 1000, as well as newton-cg solvers. After picking these hyperparameters the model was run on the test data with an accuracy of 93 %.



Classification Report:	precision	recall	f1-score	support
0	0.92	0.86	0.89	14
1	0.93	0.96	0.94	26
accuracy			0.93	40
macro avg	0.92	0.91	0.92	40
weighted avg	0.92	0.93	0.92	40

3.1.2 Ridge Regression

As expected, ridge regression showed similar results to logistic regularization, with the best alpha value for the model being 0.2 Once again, the lower alpha value indicated that there doesn't need to be much weight placed on the penalty terms of logistic classification. As shown, when running the model of test data, there was an accuracy of 90%.

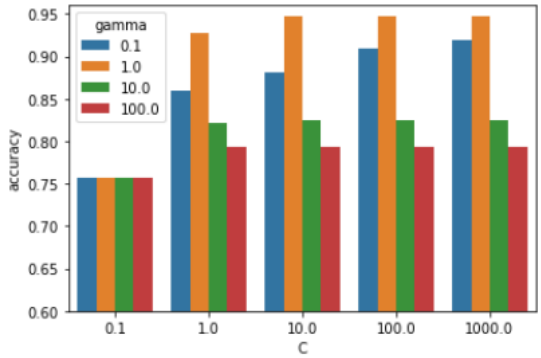


Classification Report:

	precision	recall	f1-score	support
0	1.00	0.71	0.83	14
1	0.87	1.00	0.93	26
accuracy			0.90	40
macro avg	0.93	0.86	0.88	40
weighted avg	0.91	0.90	0.90	40

3.1.3 SVM

For SVM a gamma value of 1, had the highest accuracies on training data across the board. Once that hyperparameter was picked, the focus was on finding the best C value. Plotting all the accuracies close to each other exhibits values greater than 10 doesn't improve accuracy significantly. Thus, the best model was SVM with a C value of 10 and gamma of 1, with an accuracy of 95 percent on training data and testing data.

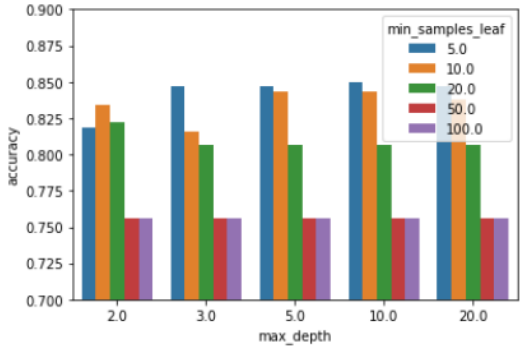


Classification Report:

	precision	recall	f1-score	support
0	0.93	0.93	0.93	14
1	0.96	0.96	0.96	26
accuracy			0.95	40
macro avg	0.95	0.95	0.95	40
weighted avg	0.95	0.95	0.95	40

3.1.4 Decision Tree

After observing the various combinations of min samples per lead and maximum depth it was found that a depth of 10 and samples of 5 resulted in the highest accuracy for our training data at 85%. This increased to 88% when running on testing data.

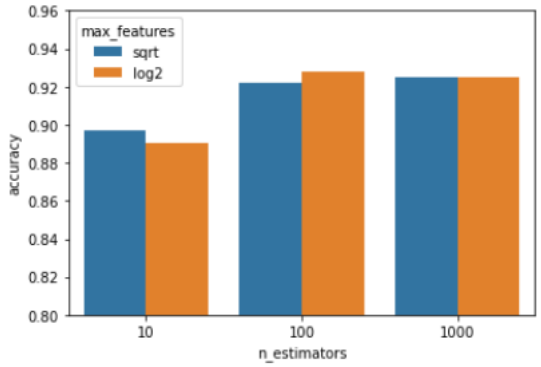


Classification Report:

	precision	recall	f1-score	support
0	0.85	0.79	0.81	14
1	0.89	0.92	0.91	26
accuracy			0.88	40
macro avg	0.87	0.85	0.86	40
weighted avg	0.87	0.88	0.87	40

3.1.5 Random Forest

As shown, the best model for random forest is one with log2 used to determine the maximum number of features, and 100 estimators used. There was a 93% accuracy on testing data.

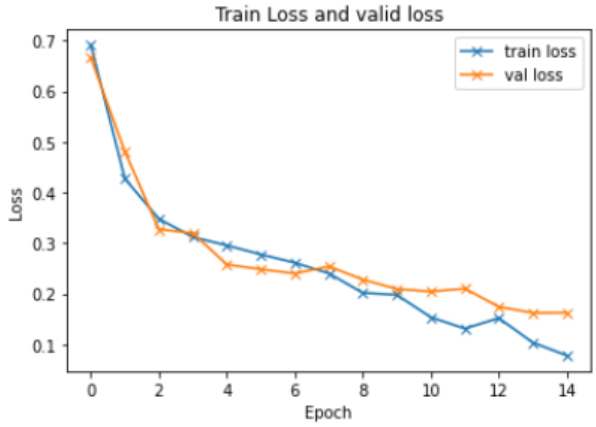


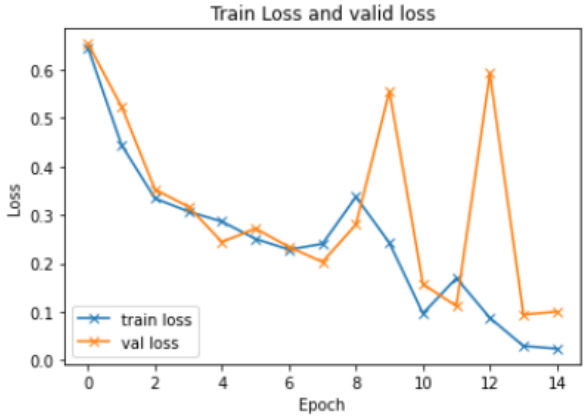
Classification Report:

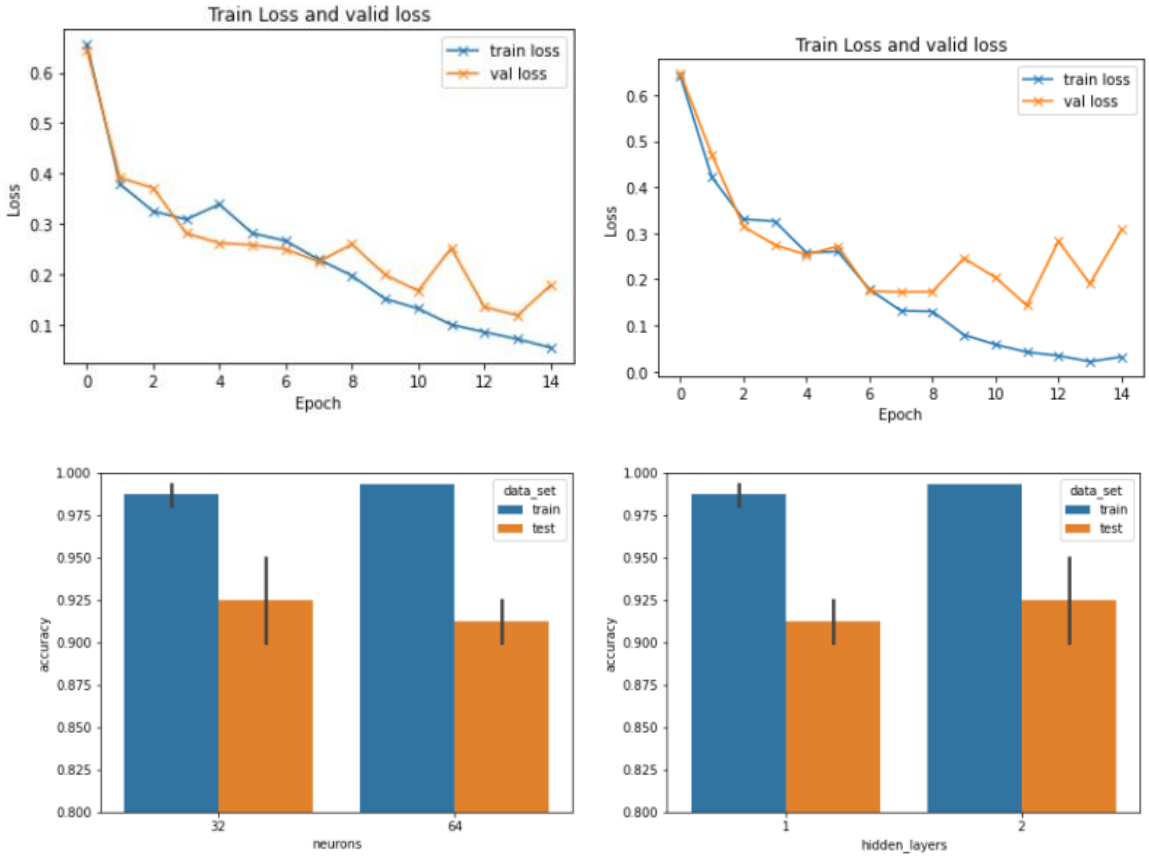
	precision	recall	f1-score	support
0	1.00	0.79	0.88	14
1	0.90	1.00	0.95	26
accuracy			0.93	40
macro avg	0.95	0.89	0.91	40
weighted avg	0.93	0.93	0.92	40

3.1.6: Feed Forward Neural Network

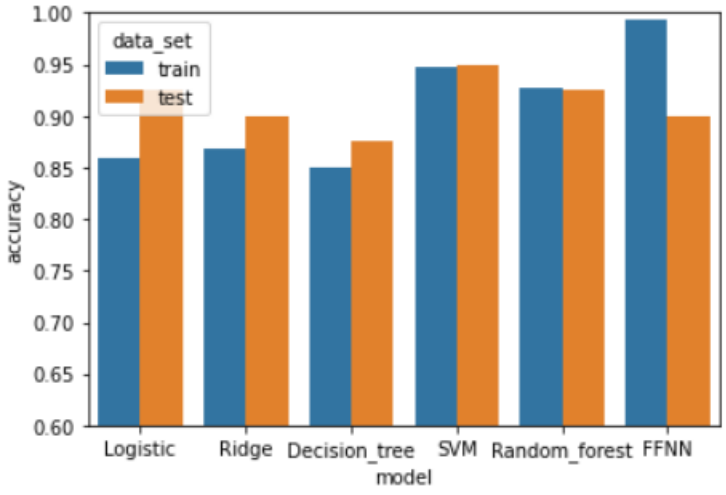
Creating a feed forward neural network for binary classification was a little more difficult than the previous classification models. The data was trained 281 epochs with one hidden layer having 32 neurons, two hidden layers with 32 neurons, 1 hidden layer with 64 neurons, and two hidden layers with 64 neurons each. The graphs display the training and validation loss across epochs with mean accuracy. The highest accuracy model on test data was with 2 hidden layers with 32 neurons at each layer. The decrease in accuracy with more hidden layers demonstrates the extra training that occurs, which is not as necessary for the simpler dataset being used.







3.2 Comparison of different methods



Once the best models were picked from the previous steps, the accuracies across the models were compared on both the testing and training data. In general, all the models performed well with accuracies at approximately 90%. In terms of training data, FFNN performed the best with an accuracy of 99%. However, this can be explained by possible overfitting, which is seen in the lower accuracy in testing data. Overall, the best model is SVM with an accuracy of 95% on both the training and testing data.

4 Conclusions

After running cross validation on each model and comparing the accuracies it's clear that there are many procedures which can be used to classify for Parkinson's disease using voice measurement data. In general, SVM worked the best, which is expected due to its method in finding a hyperplane while still accounting for misclassification errors. While FFNN performed well on testing data, it takes longer to run and isn't that accurate on testing data. This can be explained by possible overfitting. Furthermore, high accuracies across the board can be indicative of the smaller data set, which should be addressed in the future with more data included. This can also explain why lower coefficients for the penalty term were seen to perform better.

In general, these results are significant as it's a start on bettering the diagnosis for Parkinson's patients and improving patient's health, by beginning treatment measures earlier on.

5 References

[1] Alam, M. (2022, April 25). Feature selection: A comprehensive list of strategies. Medium. Retrieved March 3, 2023, from https://towardsdatascience.com/featureselection-a-comprehensive-list-of-strategies-3fecdf802b79

[2] Little, M. A., McSharry, P. E., Roberts, S. J., Costello, D. A. E., & Moroz, I. M. (2007). Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. BioMedical Engineering OnLine, 6(1), 23. https://doi.org/10.1186/1475-925x6-23

[3] Little, M., McSharry, P., Hunter, E., Spielman, J., & Ramig, L. (2008). Suitability of dysphonia measurements for telemonitoring of parkinson's disease. Nature Precedings. https://doi.org/10.1038/npre.2008.2298.1

[4] Sakar, C. O., Serbes, G., Gunduz, A., Tunc, H. C., Nizam, H., Sakar, B. E., Tutuncu, M., Aydin, T., Isenkul, M. E., & Apaydin, H. (2019). A comparative analysis of speech signal processing algorithms for parkinson's disease classification and the use of the tunable Qfactor wavelet transform. Applied Soft Computing, 74, 255–263. https://doi.org/10.1016/j.asoc.2018.10.022

[5] Speech therapy and PD. Parkinson's Foundation. (n.d.). Retrieved March 3, 2023, from https://www.parkinson.org/library/fact-sheets/speechtherapy#:~:text=Speech%20and%20Voice%20Disorders%20in%20Parkinson's&text=These%20problems%20with%20muscle%20activation,(reduced%20clarity%20of%20speec h).

[6] What is Parkinson's? Parkinson's Foundation. (n.d.). Retrieved March 3, 2023, from https://www.parkinson.org/understanding-parkinsons/what-is-parkinsons

Link to dataset: https://archive.ics.uci.edu/ml/datasets/parkinsons & https://archive.ics.uci.edu/ml/datasets/Parkinson%27s+Disease+Classification