Comparing Machine Learning Models for Predicting Epileptic Seizures

Tania Soutonglang

Illinois Institute of Technology

CS 584 Machine Learning

Professor Yan Yan

April 24, 2023

Comparing Machine Learning Models for Predicting Epileptic Seizures

Epilepsy is a neurological disorder in which brain activity becomes abnormal, causing periods of unusual behavior, sensations, and sometimes loss of awareness, but seizures are the most common form of signs of epilepsy. Continuous research has gone into correctly diagnosing epilepsy, showing how electroencephalogram (EEG) is crucial for accurate diagnosis. In this project, I will be comparing four different machine learning models to find the most effective method of classifying seizure activity from EEG signals. This project is based on Bekir Karlık and Şengül Bayrak Hayta's research, "Comparison Machine Learning Algorithms for Recognition of Epileptic Seizures in EEG," for the 2014 International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO).

Background

What is an EEG?

An EEG is a test or record of brain activity and can help detect abnormalities in brain waves through the shape of the lines frequency, and patterns of the recordings. A normal EEG is uniform in height, width, and speed. The differences occur when you're asleep and the brain starts to slow down, giving a slower and low amplitude. Abnormal EEGs have breaks interrupting the pattern found in normal EEGs (“What Does a Normal EEG Look Like? What to Look For”).

Dataset Source

The data used in this experiment, "Epileptic Seizure Recognition" came from Kaggle. The attribute information already included the background source of the data. It was originally 5 separate folders, each with 100 files representing a different participant in the study. Each participant was recorded for 23.6 seconds, giving 4,097 data points during that time. A data point

is the EEG value at that moment in time. 4,097 data points split into 23 seconds give 178 data points per second so there are 178 columns of data. 500 participants times 23 rows per person give 11,500 rows of information which were then randomly ordered ("Epileptic Seizure Recognition").

The dataset contained 5 distinct labels:

1. Recording of seizure activity.
2. They recorded the EEG from the area where the tumor was located.
3. They identified where the region of the tumor was in the brain and recorded the EEG activity from the healthy brain area.
4. The patient had their eyes closed at the time of recording the EEG signal.
5. The patient had their eyes open when recording the EEG signal.

To make classification simpler, the dataset was changed into a binary classification of "no seizure" vs. "seizure". All rows with labels 2 to 5 were given a 0 indicating "no epileptic seizure" and rows with a label of 1 were given a 1 indicating "seizure detected". Of the 500 participants, 100 of them had signs of epilepsy, meaning 2,300 rows will be labeled with a 1, and 9,200 rows will be labeled with a 0 ("Epileptic Seizure Recognition").

Methods

As previously mentioned, I will be comparing 4 different algorithms. I will be using logistic regression, support vector machine (SVM), Naive Bayes, and k-nearest neighbors (k-NN).

Logistic Regression

Logistic regression is a regression model that uses mathematics to find the relationship between two independent variables and then uses that relationship to predict the value of a binary

dependent variable (“What Is Logistic Regression?”). Most regression models use a logistic function, such as the sigmoid or Tanh functions, to predict the probability of the dependent variable.

Support Vector Machine

Support vector machines (SVM) find a hyperplane in an N-dimensional space that distinctly classifies the data points. The points are plotted on the N-dimensional space and the algorithm finds a hyperplane that gives the most significant margin between data points of both classes (Gandhi, "Support Vector Machine - Introduction to Machine Learning Algorithms"). Support vectors are data points closer to the hyperplane and influence its position and orientation.

Naive Bayes

A Naive Bayes model is a probabilistic model based on Bayes' theorem of finding the probability of A happening given that B has occurred. The theorem assumes that the features are independent of one another, meaning that the absence of one feature will not affect the likelihood of another feature occurring. Several variants of Naive Bayes models exist, such as Gaussian (which assumes that the features are normally distributed), Multinomial (which is commonly used for text classification), and Bernoulli (which is for binary data) (Gandhi, "Naive Bayes Classifier").

k-Nearest Neighbors

The k-Nearest neighbor algorithm (k-NN) uses proximity to make classifications about the grouping of an individual data point, working off the assumption that similar points can be found near one another (“What Is the K-Nearest Neighbors Algorithm?”). It uses distance

metrics, such as the Euclidean or Manhattan distance, to find the closest data points and assigns the label based on the most common label among its neighbors.

Results

Metrics

After creating and training each of the models, predictions were created. Based on the number of true positives (points correctly identified as a 1), false positives (points incorrectly identified as a 1), true negatives (points correctly identified as a 0), and false negatives (points correctly identified as a 0), the accuracy scores and F1 scores were calculated. Accuracy is a metric that measures the number of predictions that are correct over the total number of predictions made.

$$Accuracy = \frac{True\ positive}{Total\ number\ of\ predictions} \tag{1}$$

While accuracy is an important metric to evaluate models, it's good practice to also use metrics that can implement the other metrics that can be evaluated. The F1 score takes into account the number of prediction errors and looks into the types of errors that were made. It is made up of the precision and recall scores. Precision counts the correct positive outcomes out of all the positive predictions:

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{2}$$

whereas recall measures the model's ability in finding all positive cases in the data:

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negatives} \tag{3}$$

These two equations make up the F1 score, which finds the average between the precision and recall scores (Korstanje).

$$F1\ Score\ =\ 2\ *\ \frac{Precision\ *\ Recall}{Precision + Recall} \quad (3)$$

The accuracies and F1 scores are listed below:

Model	Accuracy	F1 Score
Logistic Regression	0.817826	0.183236
Support Vector Machine	0.978261	0.944934
Naive Bayes	0.963913	0.910849
k-Nearest Neighbors	0.925217	0.774278

Table 1: The accuracies and F1 scores of each model.

Based on the table we can assume that the SVM would be the most ideal model as it has the highest accuracy and F1 score, but to confirm this I will use another performance measure.

ROC Curve

Another performance measurement used is the AUC-ROC Curve (Area Under the Curve - Receiver Operating Characteristics Curve). The ROC curve measures how capable the model is in distinguishing between classes, with a higher AUC indicating better accuracy.

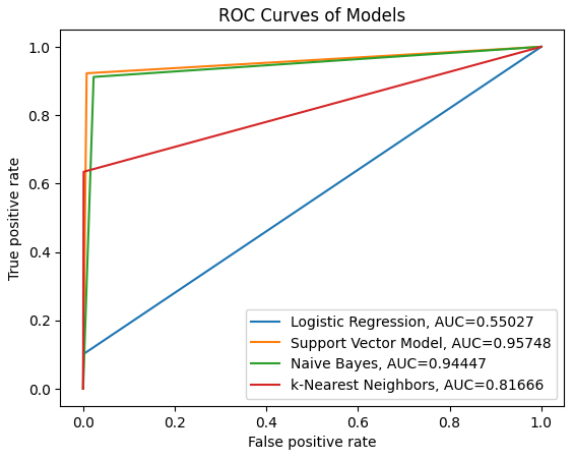

Figure 1: ROC curve of each model.

Based on the graph, the SVM model's curve and AUC are higher than the other models. This confirms the assumption that the SVM is the most ideal model for this problem.

Discussion

The aim of this project is to classify epileptic seizures using 4 supervised learning machine learning models. After training, testing, and evaluating the performance metrics, it can be concluded that the SVM is the most ideal model for this problem. It achieved the highest accuracy score, F1 score, and ROC-AUC curve, determining it will classify each input the most correctly. In the future, it would be interesting to apply the model in lab research to see it be validated in a real-world setting. It would also be helpful to apply this project to more advanced machine learning algorithms to see if those will create models with a better accuracy score.

References

Cauchi, Jonathan, et al. “Using Machine Learning to Predict Epileptic Seizures from Eeg Data.” *SSRN Electronic Journal*, Elsevier BV, July 2022. *Crossref*, doi:10.2139/ssrn.4157506.

“Epilepsy - Symptoms and Causes - Mayo Clinic.” *Mayo Clinic*, 7 Oct. 2021, https://www.mayoclinic.org/diseases-conditions/epilepsy/symptoms-causes/syc-20350093.

“Epileptic Seizure Recognition.” *Kaggle: Your Machine Learning and Data Science Community*, https://www.kaggle.com/datasets/harunshimanto/epileptic-seizure-recognition?select=Epileptic+Seizure+Recognition.csv.

Gandhi, Rohith. “Naive Bayes Classifier.” *Towards Data Science*, 5 May 2018, https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c.

---. “Support Vector Machine — Introduction to Machine Learning Algorithms.” *Towards Data Science*, 7 June 2018, https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47.

Karlık, Beki, and Şengül Bayrak Hayta. *Comparison Machine Learning Algorithms for Recognition of Epileptic Seizures in EEG* . IWBBIO, 2014.

Korstanje, Joos. “The F1 Score.” *Towards Data Science*, 31 Aug. 2021, https://towardsdatascience.com/the-f1-score-bec2bbc38aa6.

“Matplotlib.Pyplot — Matplotlib 3.5.3 Documentation.” *Matplotlib — Visualization with Python*, https://matplotlib.org/3.5.3/api/_as_gen/matplotlib.pyplot.html.

“Pandas - Python Data Analysis Library.” *Pandas - Python Data Analysis Library*, https://pandas.pydata.org/.

"Scikit-Learn: Machine Learning in Python — Scikit-Learn 1.2.2 Documentation." *Scikit-Learn: Machine Learning in Python*, https://scikit-learn.org/stable/index.html.

"What Does a Normal EEG Look Like? What to Look For." *Zeit Medical*, https://www.zeitmedical.com/post/what-does-a-normal-eeg-look-like-what-to-look-for.

"What Is Logistic Regression?" *Amazon Web Services, Inc.*, https://aws.amazon.com/what-is/logistic-regression/.

"What Is the K-Nearest Neighbors Algorithm?" *IBM - United States*, https://www.ibm.com/topics/knn.