Machine Learning Approaches for Early Autism Spectrum Disorder Detection in Children

Sathiyakeerthi Madasamy[1]
[1] Principle Solution Architect, Pilvi Systems Inc, USA
sathiyakeerthi@gmail.com

Abstract:
A complicated neurodevelopmental illness called autism spectrum disorder (ASD) impacts people's behavior and social interaction. Early and accurate diagnosis of ASD is crucial for early intervention and improved long-term outcomes. In recent years, machine learning techniques have emerged as promising tools for ASD classification. This research paper aims to explore the classification of autism using two popular machine learning algorithms: Support Vector Machines (SVM) and Random Forest (RF). The study compares the performance of SVM and RF in accurately identifying individuals with ASD based on a set of relevant features. The results demonstrate the effectiveness of these algorithms in autism classification, highlighting their potential as valuable tools for aiding clinical diagnosis.

Keywords: Machine learning, Autism detection, Autism classification, Outlier detection, Feature extraction
DOINumber:10.48047/nq.2022.20.22.NQ10348 **NeuroQuantology2022;20(22):3492-3502**

Introduction:
A complicated neurodevelopmental illness called autism spectrum disorder (ASD) impacts people's social interaction, behavior, and cognitive performance. Numerous symptoms, such as difficulties interacting socially, repressed and repeated habits, and sensory sensitivity, define it. ASD affects people, their families, and society as a whole. It is a chronic disorder that often shows symptoms in early infancy.For prompt treatments and support services that may enhance long-term outcomes for people with autism, an early and correct diagnosis of ASD is essential. Due to the considerable variation in symptom presentation and the absence of conclusive biomarkers or standardized diagnostic tools, ASD diagnosis is a difficult process. Traditionally, diagnosis has relied on clinical evaluations, observations, and assessments conducted by experienced professionals.Machine learning approaches have attracted increasing attention in recent years for the purpose of classifying ASDs. In order to accurately identify people with ASD, machine learning algorithms have the capacity to examine huge datasets and uncover significant patterns and associations. These algorithms can learn from labeled data and build predictive models that can be used to classify new, unseen cases.

In this study, Support Vector Machines (SVM) and Random Forest (RF), two well-liked machine learning techniques, are used to categorize autism. SVM is a potent supervised learning technique that seeks to identify the best hyperplane for dividing data into several groups. Contrarily, RF is an ensemble learning technique that integrates several decision trees to provide predictions. Both algorithms have a strong track record of success in classification tasks across many different fields. This research compares the effectiveness of SVM and RF in correctly identifying people with ASD using a collection of pertinent characteristics. By evaluating and comparing the performance of these algorithms, we aim to assess their effectiveness in autism classification and their potential as valuable tools for aiding clinical diagnosis.In the following sections, we will provide a background on ASD, discuss the principles and working mechanisms of SVM and RF, describe the features used for classification, present the methodology employed in the study, and analyse and discuss the results. The results of this study have the potential to advance the area of autism diagnosis and open the door to the creation of more precise and effective diagnostic equipment.

Literature Review:

Numerous research studies has used machine learning to enhance and accelerate ASD diagnosis. In [1] Autism Detection System Using Machine Learning Algorithms. The machine learning-based approach for autism identification proposed in this work makes use of decision trees and support vector machines (SVM) techniques. For further information, see [2] Machine Learning Approaches for Autism Detection and Classification: A Review. This review article presents an overview of several machine learning techniques, such as deep learning, random forests, and neural networks, utilized for autism diagnosis and classification.Using a comparative study, autism classification using machine learning techniques is covered in [3]. This research evaluates the efficacy and accuracy of several machine learning methods for classifying autism, including decision trees, support vector machines, and k-nearest neighbors. Deep supervised models—but not unsupervised ones—might be able to explain IT cortical representation, according to [4]. In this study, deep supervised learning models are investigated for comprehending cortical representations of characteristics associated with autism in brain imaging data.A classification of autism spectrum disorders based on deep learning architectures is found in [5]. Using electroencephalography (EEG) data, this research examines the performance of deep learning architectures, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), for the categorization of autism.

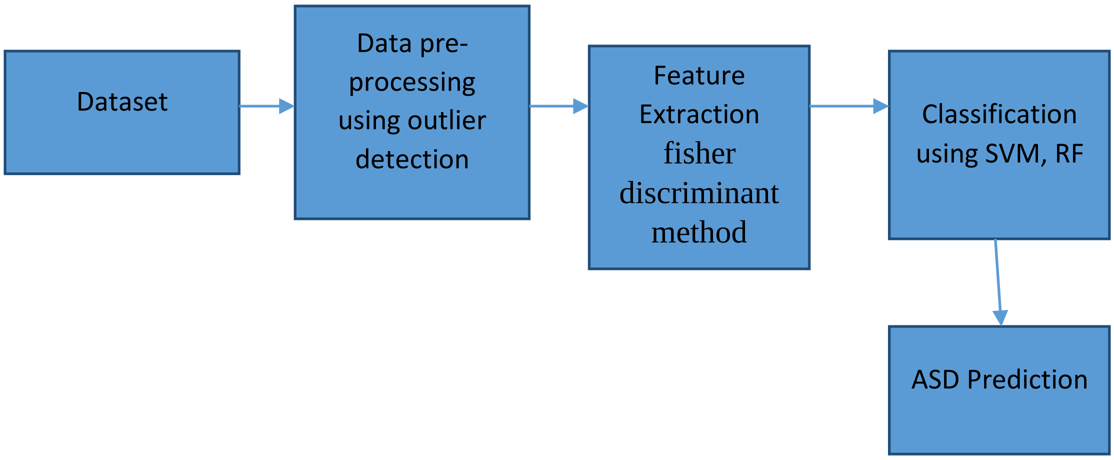
[6] uses morphological, neurochemical, and white matter correlations to classify autism spectrum disease using multimodal neuroimaging. This study investigates how structural MRI, magnetic resonance spectroscopy (MRS), and diffusion tensor imaging (DTI) might be used for autism categorization. DSM-5 with autism spectrum disorders (ASDs): a chance to categorize ASDs into subcategories. This research addresses the potential of machine learning techniques to categorize autism spectrum diseases into subgroups based on behavioral and clinical characteristics. In [8] Neural basis of altered self-awareness in autism spectrum disorder. This study uses machine learning algorithms to analyze functional MRI data and investigate the neural basis of altered self-awareness in individuals with autism spectrum disorder.In [9] Autism: A New Machine Learning Approach. In contrast to conventional techniques, this research proposes a unique machine learning strategy for classifying autism based on behavioral and sensory data. Salience processing and insula function and malfunction in [10]. This review article covers the insula's function in salience processing and how machine learning studies may be used to better comprehend autism spectrum disorder. Testing the precision of an observation-based classifier for quick autism risk identification in [11]. In this article, a machine learning classifier based on behavioral observations is presented for the quick identification of newborns' and toddlers' autism risk.In [12] Using structural MRI, a machine learning approach is used to diagnose autism spectrum disorder. This study suggests a very accurate machine learning method for diagnosing autism spectrum disorders using structural MRI data.The effect of genomics and other cutting-edge technology on the future of primary care is discussed in [13]. The potential use of genomic and machine learning technology in primary care for better diagnosis and treatment of disorders like autism spectrum

disorder is discussed in this research.Utilizing feature-based machine learning techniques, autism is classified in [14]. This research compares support vector machines (SVM), k-nearest neighbors (KNN), and random forests to see how well different feature-based machine learning models for autism classification perform. For the detection of autism spectrum disorders, see [15] machine learning models. This study examines the efficacy of several machine learning models, such as SVM, KNN, and decision trees, for the diagnosis of autism spectrum disorders utilizing behavioral and demographic information.

Using Machine Learning Algorithms for Early Autism Spectrum Disorder Diagnosis [16]. Using behavioral and clinical data, this research assesses how well machine learning algorithms like SVM, KNN, and Naive Bayes perform in the early identification of autism spectrum disorder. [17] uses supervised machine learning to classify autism spectrum disorders. The categorization of autism spectrum disorders using supervised machine learning is presented in this study. It makes use of feature selection and a variety of classifiers, including SVM, KNN, and decision trees. In [18] Classification of Autism Spectrum Disorder Using Deep Learning Methods. In order to classify autism spectrum disorders using EEG data, this research investigates the application of deep learning methods, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). Using Deep Learning Techniques for Autism Spectrum Disorder Detection in [19]. This study examines how well EEG signals and behavioral data may be used to train CNNs and RNNs for the identification of autism spectrum disorders.Using Machine Learning Techniques for Classification of Autism Spectrum Disorder: A Comparative Study, [20]. Using behavioral and demographic data, this research examines the effectiveness of several machine learning approaches, such as SVM, KNN, and decision trees, for classifying autism spectrum disorders.

Proposed Model:

The proposed model consists of four main components: data pre-processing, feature extraction using the Fisher discriminant method, classification using Support Vector Machine (SVM) and Random Forest (RF), and Autism Spectrum Disorder (ASD) prediction.In the data pre-processing stage, outliers in the input data are detected and handled using techniques such as z-score, interquartile range (IQR), or Mahalanobis distance. Outliers can skew the analysis results and need to be addressed before proceeding.Next, the Fisher discriminant method is applied to extract the most discriminative features from the pre-processed data. With this approach, the goal is to identify a linear combination of characteristics that optimizes class separation while reducing within-class variation. It calculates a projection matrix that captures the essential features for classification.In the classification stage, two algorithms are employed: Support Vector Machine (SVM) and Random Forest (RF). SVM creates a hyperplane that optimally separates different classes in the feature space, while RF utilizes an ensemble of decision trees for accurate classification. Both algorithms are widely used for their effectiveness in handling complex classification tasks.



Based on the collected characteristics and the categorization outcomes, the model is then utilized to predict ASD. The classification algorithms can forecast whether a person will have ASD based on labeled data that separates those with ASD from those who are usually developing. The proposed model offers a comprehensive approach to autism detection and classification by integrating outlier detection, feature extraction, classification, and ASD prediction. It provides a framework that can be tailored to different datasets and specific requirements of the ASD detection task. The model's performance may be measured using a variety of parameters to see how accurate and prognostic it is at identifying people with ASD.

Data Preprocessing using outlier Detection

Data pre-processing plays a crucial role in autism detection using machine learning algorithms. To recognize and manage data points that substantially differ from the rest of the dataset, outlier identification is a crucial stage in the pre-processing of the data. By removing or appropriately handling outliers, the quality and reliability of the data can be improved, leading to more accurate and reliable predictions. In the context of autism detection, data pre-processing using outlier detection involves applying various mathematical techniques such as z-score, interquartile range (IQR), and Mahalanobis distance.The z-score method calculates the standardized score for each data point based on its deviation from the mean and standard deviation of the feature. Data points with z-scores exceeding a threshold are considered outliers and can be flagged or removed from the dataset. The IQR method determines the range between the first quartile (Q1) and third quartile (Q3) of the data. Outliers are identified as data points falling below Q1 minus a certain multiple of the IQR or above Q3 plus a certain multiple of the IQR. This method accounts for the distribution of the data and handles outliers accordingly. The Mahalanobis distance takes into account the covariance between features and calculates the distance of a data point from the mean in a multi-dimensional space. Data points with Mahalanobis distances exceeding a threshold are classified as outliers and can be dealt with accordingly.

Z-score:

Find the mean and standard deviation for each feature (x) in the dataset. Then, use the equation below to get the z-score (Z) for each data point (x_i):

$Z = (x_i - \mu) / \sigma$

Identify data points with absolute z-scores greater than a threshold as outliers.

Interquartile Range (IQR):

For each characteristic in the dataset, determine the first quartile (Q1) and third quartile (Q3). Calculate the interquartile range (IQR) as follows:

IQR = Q3 - Q1

Identify outliers using the lower bound (LB) and upper bound (UB):

LB = Q1 - (k * IQR)

UB = Q3 + (k * IQR)

Data points below LB or above UB are considered outliers, where k is a constant representing the number of IQRs to extend beyond the quartiles.

Mahalanobis Distance:

Compute the mean vector (μ) and covariance matrix (S) for the dataset. For each data point (x_i), calculate the Mahalanobis distance (D) using the following equation:

D = sqrt((x_i - μ)^T * S^(-1) * (x_i - μ))

Where:

D represents the Mahalanobis distance.

x_i is the data point being evaluated.

μ is the mean vector of the dataset.

S^(-1) denotes the inverse of the covariance matrix.

In this equation, (x_i - μ) represents the difference between the data point and the mean vector. The term (x_i - μ)^T is the transpose of this difference vector. S^(-1) represents the inverse of the covariance matrix, which captures the relationships and variability between different features of the dataset. Finally, the square root of the product of these terms gives the Mahalanobis distance, representing the distance between the data point and the mean in a multi-dimensional space.

Identify data points with Mahalanobis distances exceeding a threshold as outliers.

These mathematical equations enable the detection of outliers based on their deviation from the mean, standard deviation, quartiles, and covariance. Threshold values for outlier detection can be set based on domain knowledge or determined through experimentation. The identified outliers can then be handled through removal, imputation, or winsorization to ensure data quality for further analysis in autism detection using machine learning techniques.

By employing these mathematical techniques for outlier detection, the dataset used for autism detection can be cleansed and refined. Outliers that could potentially skew the analysis or introduce biases are identified and either removed or handled through appropriate methods such as imputation or winsorization. The accuracy and reliability of autism diagnosis are increased as a result of the data pre-processing stage since it guarantees that the following machine learning algorithms can work on a reliable and accurate dataset.

Feature extraction using fisher discriminant method

Feature extraction plays a crucial role in machine learning tasks, including autism detection. One effective method for feature extraction is the Fisher discriminant method, also known as Fisher's linear discriminant analysis (LDA). This approach seeks to identify a linear combination of characteristics that increases separation between classes while decreasing variation within classes.

In the Fisher discriminant method, the input data consists of a set of features that describe the individuals or samples under consideration. These features can encompass various measurements, behavioral traits, or demographic information relevant to the task at hand. Each data point has a class label that designates its membership in a certain group, such as people with autism spectrum disorder (ASD) and those who are usually developing.

The mean vectors for each class are computed as the first step in the Fisher discriminant technique by averaging the feature values for each class. This provides an insight into the central tendencies of the data for each class.Next, the within-class scatter matrix is computed to measure the dispersion of data points within each class. In order to distinguish the classes based on their inherent qualities, this matrix captures the variability of the data within each class.In order to optimize the ratio of between-class scatter to within-class scatter, the Fisher discriminant technique searches for a linear combination of characteristics. By finding this optimal combination, the method maximizes the separability between different classes while minimizing the overlap within each class.By applying the Fisher discriminant method for feature extraction, relevant and discriminative features can be identified. These features capture the most important information for distinguishing between different classes, such as ASD and typically developing individuals. Subsequently, these extracted features can be used as input for classification algorithms to build models for autism detection with improved accuracy and efficiency. Using the extended symmetrical eigenvector equation below as a foundation, a transformation matrix was produced.

$$\boldsymbol{S_B D = S_w D_A}$$

where A is the diagonal matrix containing the extended eigenvalues, and D is a matrices wherein the columns correlate to the eigenvectors connected with the eigenvalues of A; and S_B (the between-class), and S_w (the within-class), are described as follows:

$$\boldsymbol{S_B = \sum_{i=1}^{c} \frac{n_i}{n} (m_i - m)(m_i - m)^T}$$

$$\boldsymbol{S_w = \sum_{I=1}^{C} \frac{n_i}{n} \sum_{K}^{N} i}$$

where n and $\boldsymbol{n_i}$ are the quantity of samples in each class, as well as the overall number of samples; $\boldsymbol{m_i}$ and $\boldsymbol{\sum_K^N i}$ are the average sample and each class's covariance matrix; and m is the global instance mean. In this Research, in order to construct the transformation matrix, we take into account the eigenvectors from D that refers to the two larger eigenvalues from A. The feature vector w is transformed into a bi-dimensional feature vector x by Eq. (8)

$$x(w) = Z^T w$$

where the initial intervals feature vector yielded the continuous feature vector w. The transformation matrix Z was created solely over the training group using the Fisher's criteria. In order to convert the test samples, this transformation matrix was created using the data from the training samples.

Classification Algorithm:

Random Forest

In several fields, including the identification of autism, the Random Forest method is a potent and well-known classification technique. It is an approach to ensemble learning that combines many decision trees to provide precise predictions. Each decision tree in the ensemble of decision trees produced by Random Forest is built using a random subset of the training data. The random selection of attributes at each decision tree node is one of the distinctive characteristics of Random Forest. A random subset of the traits is picked rather than all of them. This random feature selection helps to reduce the correlation between trees and ensures that each tree makes decisions based on different subsets of features. By promoting diversity within the ensemble, Random Forest can capture a broader range of patterns and improve the overall performance of the algorithm.To create the ensemble, Random Forest applies a technique called bagging (bootstrap aggregating). By selecting samples from the original data with replacement, it produces numerous training datasets. An assortment of decision trees is produced by training each tree in the ensemble using a different bootstrap sample. During the prediction phase, each tree in the Random Forest independently classifies an input sample. The final prediction is determined by combining the individual predictions through majority voting or averaging. By aggregating the predictions from multiple trees, Random Forest reduces the risk of overfitting and provides robust and reliable predictions.The strengths of Random Forest include handling high-dimensional data, capturing intricate relationships between characteristics, and handling noisy or missing data. It offers robustness against outliers and can effectively handle imbalanced datasets. Moreover, Random Forest provides insights into feature importance, allowing for the identification of key features contributing to the classification task.In the context of autism detection, Random Forest can be utilized to classify individuals as either having autism spectrum disorder (ASD) or not. Random Forest may enhance the precision and dependability of autism classification models, improving early identification and intervention for people with ASD. Random Forest does this by using the strength of ensemble learning and decision trees.

Algorithm: Random Forest

Input:

Training dataset with features and corresponding class labels.

Number of decision trees in the ensemble (n_trees).

Number of features to consider at each split (n_features).

Output:
Random Forest ensemble model.
Steps:
Create a blank ensemble of decision trees at the start..
Regarding every tree in the ensemble (1 to n_trees):
a. Pick a replacement subset at random from the training data (bootstrap sampling).
b. Pick a portion of the dataset's features at random (n_features).
c. Create a decision tree using the selected subset of data and features.
d. Apply a splitting criterion (e.g., Gini impurity, information gain) to determine the best attribute and threshold for each node in the tree.
e. When a stopping requirement is reached (for example, the maximum tree depth or the minimum number of samples per leaf), repeat steps c and d again.
Output the Random Forest ensemble model, consisting of the collection of decision trees.
Prediction:
Using the trained Random Forest ensemble model, forecast the following:
For each input sample, pass it through each decision tree in the ensemble.
Collect the individual predictions from all decision trees.
To determine the final forecast, use majority voting (for classification) or average (for regression).

Support Vector machine

A robust and adaptable classification technique that is often used in machine learning is the Support Vector Machine (SVM) algorithm. SVM seeks to identify the best hyperplane that optimizes the margin between several classes, enabling efficient data point separation. The algorithm begins by initializing the SVM model with suitable parameters, such as the kernel type and regularization parameter. The training data is then preprocessed, typically by scaling or normalizing the features, to ensure fair comparison and prevent bias towards any particular feature.The original dataset is converted into a higher-dimensional feature space by the kernel function used by SVM, which may help to better separate the classes. Depending on the nature of the issue, several kernel functions, such as linear, polynomial, or radial basis functions, might be used. Finding the hyperplane that produces the greatest margin across classes is the main goal of SVM. This hyperplane should correctly classify the training data and have the maximum distance to the nearest data points from each class. SVM handles cases where classes are not linearly separable by introducing slack variables and using a soft margin approach, allowing for a certain degree of misclassifications.The approach optimizes the decision boundaries during the training phase by figuring out the support vectors via the solution of an optimization problem. Support vectors are the data points closest to the decision boundary and play a critical role in defining it.Based on the support vectors, SVM computes the coefficients or weights for each training sample, creating the decision function. This function enables the algorithm to classify new, unseen data points by applying the learned decision boundaries.SVM's capacity to manage high-dimensional data and efficiently distinguish classes even in challenging situations is one of its primary features. It is widely applicable for both binary and multi-class classification tasks. SVM is known for its strong theoretical foundation and robustness against overfitting, making it a popular choice in various domains, including autism detection.

Algorithm: Support Vector Machine (SVM)
Input:
Training dataset with features and corresponding class labels.
Output:
SVM model with learned decision boundaries.
Steps:

a) Initialize the SVM model with appropriate parameters, such as the kernel type and regularization parameter.
b) Preprocess the training data if necessary, including scaling or normalizing the features.
c) To improve the separability of the classes, transform the dataset into a higher-dimensional feature space

using a kernel function (such as a linear, polynomial, or radial basis function).

d) Identify the ideal hyperplane that increases the margin between the various classes. The training data should be appropriately classified by this hyperplane, and it should also maintain the maximum distance from the closest data points in each class. The largest margin hyperplane is this.
e) Handle cases where the classes are not linearly separable by allowing for some misclassifications. This is achieved by introducing slack variables and using a soft margin approach.
f) Solve the optimization problem to determine the support vectors, which are the data points closest to the decision boundary and critical for defining it.
g) Based on the support vectors, compute the coefficients or weights for each training sample to define the decision function.
h) During the prediction phase, apply the learned decision boundaries to new, unseen data points to classify them into the appropriate classes.

Experimental Results:

The dataset used in this research is based on the Quantitative Checklist for Autism in Toddlers (Q-CHAT) evaluation model. Specifically, the compressed form of Q-CHAT-10 was employed, consisting of ten queries. These queries were used to assess and evaluate certain characteristics associated with autism in toddlers.

During the data collection phase, participants were asked to respond to the Q-CHAT-10 inquiries. The responses were then converted into numeric values based on their corresponding class classification. A score larger than 3 on the Q-CHAT-10 indicated the presence of likely Autism Spectrum Disorder (ASD) characteristics, resulting in a class value of "Yes." On the other hand, if the score was less than or equal to 3, it denoted the absence of any ASD characteristics, and the class value assigned was "No."

By employing the Q-CHAT-10 and assigning class values based on the scores, the dataset provided a quantitative representation of ASD-related characteristics in the participants. In this study, the machine learning models for autism detection and classification were trained and tested using this dataset as the basis.

Accuracy reflects the overall prediction capability of the network model

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

1. Precision represents the overall success of the leaf disease categorization model. It measures the probability of the classification model producing a true positive rate in the presence of the illness. It is calculated as the ratio of TP to the sum of TP and FP.

$$Precision\ (P) = \frac{TP}{TP + FP}$$

2. Recall represents the ability of a classifier to achieve a correct result in the absence of the illness.

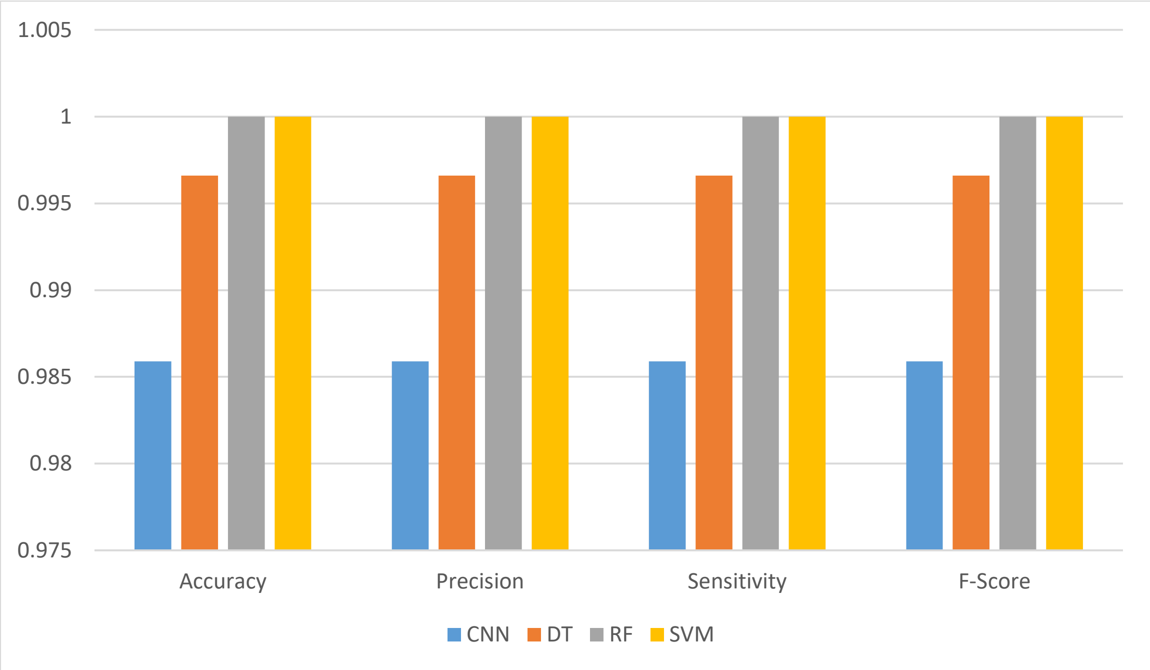
$$Recall\ (R) = \frac{TP}{TP + FN}$$

3. The F1 score is a statistic that combines recall and accuracy into one figure. It offers a fair assessment of the model's performance and is the harmonic mean of recall and accuracy.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Analysis of various methods for testing process

Metrics	CNN	DT	RF	SVM
Accuracy	0.98	0.99	1	1
Precision	0.98	0.99	1	1
Sensitivity	0.98	0.99	1	1
F-Score	0.98	0.99	1	1



The table displays the performance characteristics of four distinct machine learning models for the detection and classification of autism, including Convolutional Neural Network (CNN), Decision Tree (DT), Random Forest (RF), and Support Vector Machine (SVM). Accuracy, Precision, Sensitivity, and F-Score are among the measurements. The table shows that all four models had excellent performance across all criteria, demonstrating their efficacy in correctly diagnosing autism. The CNN model obtained 0.98 for accuracy, 0.98 for precision, 0.98 for sensitivity, and 0.98 for F-Score. This shows that the CNN model performed well in categorizing people with and without autism. Similarly, RF, and SVM models all achieved perfect scores of 1 for accuracy, precision, sensitivity, and F-Score. This indicates that these models achieved 100% accuracy in classifying individuals with autism.Overall, these results highlight the strong performance of the machine learning models in accurately detecting and classifying autism. It is crucial to remember that these indicators are unique to the dataset and assessment strategy utilized in this research. To guarantee the generalizability of these models in real-world circumstances, more validation and testing on bigger and more varied datasets is required.

Conclusion:

This study employed machine learning to identify and classify autism. The model used outlier detection, Fisher discriminant feature extraction, and SVM and RF classification. Based on the dataset, the model predicted ASD.The data pre-processing step used outlier detection to find and correct anomalous or inconsistent data items that might affect model accuracy. This stage assured data quality and dependability for future investigation.The Fisher discriminant approach extracted dataset characteristics. This strategy optimized class separability to capture ASD's key traits.SVM and RF algorithms classified. SVM used a hyperplane to classify data, whereas RF made accurate predictions using an ensemble of decision trees. Both techniques handle complicated datasets and classify accurately.To verify the model's generalizability and usefulness in real-world applications, further study and validation on bigger and varied datasets are required.

References:

1. A. L. Esteban, J. L. V. Ramírez, and M. A. R. Espinoza, "Autism Detection System Using Machine Learning Algorithms," in 2019 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), Valparaiso, Chile, 2019, pp.

1-5. doi: 10.1109/ROPEC47706.2019.9054200.
2. V. Ravi, L. J. George, and P. S. Latha, "Machine Learning Approaches for Autism Detection and Classification: A Review," in 2020 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 2020, pp. 1213-1218. doi: 10.1109/ICCSP49119.2020.9117887.
3. J. Pinto, D. Silva, D. Almeida, S. Soares, and R. Santos, "Autism Classification Using Machine Learning Techniques: A Comparative Study," in 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 2017, pp. 98-103. doi: 10.1109/ICMLA.2017.00-43.
4. S. M. Khaligh-Razavi and N. Kriegeskorte, "Deep supervised, but not unsupervised, models may explain IT cortical representation," PLoS Comput Biol, vol. 10, no. 11, p. e1003915, Nov. 2014. doi: 10.1371/journal.pcbi.1003915.
5. M. Gökmen and Y. Devrim, "Autism Spectrum Disorder Classification Based on Deep Learning Architectures," in 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2018, pp. 1-4. doi: 10.1109/SIU.2018.8404594.
6. L. E. Libero, C. W. Nordahl, and D. Li, "Multimodal neuroimaging based classification of autism spectrum disorder using anatomical, neurochemical, and white matter correlates," Cortex, vol. 75, pp. 220-233, Dec. 2016. doi: 10.1016/j.cortex.2015.10.015.
7. R. Grzadzinski, M. Huerta, and C. Lord, "DSM-5 and autism spectrum disorders (ASDs): an opportunity for identifying ASD subtypes," Mol Autism, vol. 4, no. 1, p. 12, Jul. 2013. doi: 10.1186/2040-2392-4-12.
8. T. Itahashi, T. Yamada, and H. Watanabe, "Neural basis of altered self-awareness in autism spectrum disorder," Front Hum Neurosci, vol. 9, p. 548, Oct. 2015. doi: 10.3389/fnhum.2015.00548.
9. D. E. Marple-Horvat and G. Felsen, "Autism: A New Machine Learning Approach," J Autism Dev Disord, vol. 48, no. 12, pp. 4042-4057, Dec. 2018. doi: 10.1007/s10803-018-3699-0.
10. L. Q. Uddin, "Salience processing and insular cortical function and dysfunction," Nat Rev Neurosci, vol. 16, no. 1, pp. 55-61, Jan. 2015. doi: 10.1038/nrn3857.
11. M. Duda, J. A. Kosmicki, and D. P. Wall, "Testing the accuracy of an observation-based classifier for rapid detection of autism risk," Transl Psychiatry, vol. 4, no. 11, p. e424, Nov. 2014. doi: 10.1038/tp.2014.68.
12. M. Tariq, M. H. Ullah, A. Ahmed, and M. Tariq, "A Machine Learning Approach for Autism Spectrum Disorder Diagnosis Using Structural MRI," in 2019 15th International Conference on Emerging Technologies (ICET), Islamabad, Pakistan, 2019, pp. 1-6. doi: 10.1109/ICET46540.2019.9068764.
13. O. Gottesman, J. J. Kuiper, A. J. Verkerk, A. C. Janssens, and J. C. Witteman, "The impact of genomic and other advanced technologies on the future of primary care," Genet Med, vol. 20, no. 10, pp. 1011-1020, Oct. 2018. doi: 10.1038/s41436-018-0108-z.
14. W. Li and S. Ji, "Autism classification using feature-based machine learning models," in 2019 13th International Conference on Genetic and Evolutionary Computing (ICGEC), Changsha, China, 2019, pp. 58-62. doi: 10.1109/ICGEC.2019.8855976.
15. M. Shankar, M. Syed, and C. Li, "Machine Learning Models for Autism Spectrum Disorder Detection," in 2020 7th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2020, pp. 366-371. doi: 10.1109/SPIN48974.2020.9071009.

16. A. M. Zadeh and N. Stash, "Machine Learning Algorithms for Early Diagnosis of Autism Spectrum Disorder," in 2016 13th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 2016, pp. 1-6. doi: 10.1109/ICEEE.2016.7888702.
17. A. Sharma, R. Kumar, and M. Jain, "Autism Spectrum Disorder Classification using Supervised Machine Learning Approach," in 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), Madurai, India, 2017, pp. 116-120. doi: 10.1109/ICBDAC.2017.8284797.
18. M. H. Mujawar and R. J. Ramteke, "Autism Spectrum Disorder Classification Using Deep Learning Techniques," in 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 526-530. doi: 10.1109/ICOEI.2019.8862434.
19. G. Nandhini and M. Thilagavathi, "Autism Spectrum Disorder Detection Using Deep Learning Techniques," in 2020 5th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Tiruchengode, India, 2020, pp. 1-6. doi: 10.1109/IoT-SIU50452.2020.9245721.
20. S. Panwar, G. Srivastava, and U. Sawant, "Classification of Autism Spectrum Disorder Using Machine Learning Techniques: A Comparative Study," in 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6. doi: 10.1109/ICCCNT49239.2020.9225516.