Introduction

Background

Autism Spectrum Disorder (ASD) is a "complex neurodevelopmental-condition characterized by persistent challenges in social interaction, communication, and restricted (or repetitive) activities" (WHO, 2013). The global prevalence of ASD is estimated at approximately one in 160 individuals(Elsabbagh et al., 2012), and about one in every 100 children is diagnosed with ASD every year(Zeidan et al., 2022).

People with ASD, and their families face, numerous challenges that significantly impact their quality of life. ASD is characterized by difficulties in 'social interaction, communication, and atypical behaviors', which can vary widely in severity. These challenges often cause social isolation, stigma, and misunderstanding in communities, adding to the emotional strain on families(WHO, 2013). Communication difficulties in autistic people can lead to frustration and behavior problems, impacting relationships and education(APA, 2019).

Furthermore, there is a lack of support for adults with ASD, leading to poor outcomes in employment, independent living, and overall well-being. The transition from childhood to adulthood for an autistic person is said to be especially difficult(WHO, 2013). Multiple co-occurring mental-health-conditions such as depression, ADHD and anxiety are common and frequently undertreated, compounding the difficulties faced by those with ASD(Lever & Geurts, 2016).

The burden on families is immense, as they often struggle with emotional and financial pressures, particularly when trying to access costly and time-consuming specialized services (Autism Speaks, 2017).

Problem Statement

ASD typically manifests in early childhood and continues throughout life, with symptoms and severity varying widely among individuals(Allely, 2022) . While the exact causes remain unknown, some researches have suggested a combination of genetic-and-environmental-factors contribute to its development (National Institute of Mental Health (NIMH), 2024). Early diagnosis and intervention are crucial for improving outcomes(WHO, 2013). However, many individuals, mostly in developing and under-developed countries, face significant barriers to accessing proper care and support. As awareness grows, there is an increasing focus on developing comprehensive, coordinated approaches to properly and effectively handle the complex needs of individuals with ASD across their lifespan.

Traditional diagnostic approaches for ASD are often time-consuming, expensive, and can lead to delayed diagnosis(Oma et al., 2019). Despite increased awareness and research efforts, there remains a significant gap in the provision of comprehensive, accessible, and

effective services for individuals with this disease across the lifespan, particularly in low-resource settings. In recent years, the use of machine learning and deep learning techniques have shown promise in developing more efficient and accurate methods. And yet, due to the heterogeneity of symptoms (and effects) of this disease, the world is yet to see a technological (or medical) miracle (meaning solution) that is applicable in identifying majority, if not all, cases of Autism Spectrum Disorder.

Prevalent Assessment Approaches

Behavioral Assessments

Several studies have utilized behavioral assessment tools as input features for ASD prediction models. The Autism Quotient (AQ) questionnaire and its variants have been particularly popular:

- **AQ-10 Screening Tool**: (Thabtah & Peebles, 2020) developed models based on the AQ-10, which includes questions related to social skills, attention switching, communication, and imagination. This brief screening tool has shown promise in efficiently capturing key ASD traits.
- **Expanded AQ Dataset**:(Narala et al., 2023) used an expanded dataset containing 2,940 images with AQ scores and other behavioral features. This larger dataset allowed for more robust model training and evaluation.
- **Behavioral and Demographic Features**: (Hemu et al., 2022)incorporated a wide range of behavioral features alongside demographic information, including age, gender, and ethnicity. This study highlighted the importance of considering multiple factors in ASD prediction.

Performance on Behavior Data

Several studies have reported high accuracies when using behavioral and demographic features:

- (Thabtah & Peebles, 2020)achieved **92.26% accuracy using a Random Forest** model on AQ-10 data.
- (Hemu et al., 2022)reported **97.70% accuracy with both Random Forest and XGBoost** classifiers on their behavioral dataset.
- (Naik et al., 2023)achieved accuracies **above 90% with various ML algorithms, including Random Forest, XGBoost, and SVM**, on their behavioral dataset.

Neuro-Imaging Data

Neuroimaging, particularly 'functional magnetic resonance imaging (fMRI)', has been a credible and most preferred source of information for ASD prediction models:

- **Functional Connectivity**: (Sun et al., 2023) utilized 'whole-brain functional connectivity' matrices derived from resting-state-fMRI to classify ASD and typically developing controls. They extracted 6670 features representing functional connections between 116 brain regions. They found that their ensemble approach performed well in predicting ASD.
- **Brain Network Analysis**: (Li et al., 2018) constructed functional brain networks from fMRI data to distinguish individuals with ASD. Their approach focused on identifying atypical connectivity patterns associated with ASD.

Performance on Neuro-imaging Data

Studies using neuroimaging data typically report lower but still promising accuracies:

- (Sun et al., 2023) achieved **83.42% accuracy using a Decision Tree** classifier on functional connectivity features derived from fMRI data.
- (Li et al., 2018) reported **85.3% accuracy** in classifying ASD and control subjects using their Deep Belief Network approach on fMRI data.

Facial Image Analysis

Recent studies have explored the potential of facial image analysis for ASD prediction:

- **Facial Features**: (Khosla et al., 2021) used deep learning models to analyze facial images of children for ASD detection. Their approach was based on the hypothesis that subtle facial features might be indicative of ASD.
- **Facial Expressions**:(Jaffar & Abdulbaqi, 2022) focused on facial expression recognition in static images of children with autism. Their work highlighted the potential of analyzing emotional expressions for ASD screening.

Performance on Facial Image Data

Facial image analysis has shown potential for ASD prediction:

- (Khosla et al., 2021) achieved **87% testing accuracy using their MobileNet**-based model on facial images.
- (Narala et al., 2023) reported that their EfficientNet model outperformed ResNet50, achieving higher accuracy in detecting ASD from facial images.
- (Jaffar & Abdulbaqi, 2022) CNN-based facial expression recognition model achieved **97.48% accuracy during training and 92.66% accuracy during validation**.

Multimodal Approaches

Some researchers have combined multiple data types to create more comprehensive ASD prediction models:

- **Behavioral and Neuroimaging Data**: (Akter et al., 2019) developed models using a combination of behavioral features, demographic information, and fMRI data. This multimodal approach aimed to capture a more complete picture of ASD-related traits and brain function.
- **Speech and Behavioral Data**: (Sadiq et al., 2019) analyzed both speech patterns and behavioral assessments to predict ASD. Their work demonstrated the potential of integrating different types of behavioral data for improved prediction accuracy.

In the next section, we explore different AI/ML approaches designed and used to predict and diagnose ASD, with a focus on different data types, algorithms and their performance. The objective of this study is to identify the most lightweight system that can work well for a resource limited setting.

Machine Learning Techniques for ASD Prediction

Several studies have employed various ML (machine-learning) algorithms to predict ASD based on behavioral, genetic, and neuroimaging-data. (Hemu et al., 2022) compared multiple machine-learning models, including Random Forest (RF), XGBoost, Decision Tree, and Logistic Regression, for ASD prediction among children. Their study found **that RF and XGBoost performed best, achieving 97.70% accuracy**. The researchers also used SHAP values to identify the most important features contributing to ASD prediction.

Random Forest approach has shown strong performance across multiple other studies too (Naik et al., 2023; Sethi et al., 2024). **Its ability to handle non-linear relationships and provide feature important rankings makes it a valuable approach**.

(Sun et al., 2023)utilized an ensemble machine learning approach, combining multiple classifiers such as 'K-Nearest Neighbors (KNN)', 'Naïve Bayes (NB)', 'Random Forest (RF)', 'Gradient Boosting Decision Tree (GBDT)', and 'XGBoost'. They employed **a stacking strategy with Logistic Regression as the meta-model**. This ensemble approach demonstrated improved performance compared to individual classifiers, achieving higher accuracy, sensitivity, and specificity in ASD prediction.

Despite it simplicity, **logistic regression has proven effective in several studies** as well(Hemu et al., 2022; Naik et al., 2023; Sethi et al., 2024), especially when combined with careful feature selection.

Deep Learning Techniques for ASD Prediction

Deep learning techniques, 'Convolutional Neural Networks (CNNs)' in particular, have shown significant potential in ASD prediction, especially when working with image data such

as facial images or brain scans. (Khosla et al., 2021) utilized deep-learning-models pre-trained on ImageNet to classify facial-images of children as either healthy or potentially autistic. The study found that the **MobileNet model achieved a maximum of 87%** testing accuracy in distinguishing between ASD and non-ASD children.

ResNet50, a deep residual network architecture, has been widely used in various image classification tasks due to its ability to train very deep networks effectively. (Narala et al., 2023) compared the performance of EfficientNet and ResNet50 models in detecting ASD from facial images. Their study found that both models were effective in detecting differences, with EfficientNet exhibiting higher accuracy and parameter efficiency compared to the ResNet50 model. **ResNet50's capabilities in extracting complex features from images make it a promising candidate for analyzing facial images** or neuroimaging data in ASD prediction tasks(Venkata Sai Krishna Narala et al., 2023).

(Jaffar & Abdulbaqi, 2022) developed a custom CNN model for facial expression recognition in children with Autism. Their model achieved high accuracy in distinguishing different emotional expressions. (Li et al., 2018) used a deep-belief-network to analyze fMRidata for ASD classification. The approach demonstrated the potential of deep learning in capturing complex patterns in neuro-imaging data.

The mixture of deep learning for 'feature-extraction' and machine learning for 'classification' has shown promising results. The study by (Sun et al., 2023) demonstrated **that using a deep learning model for feature extraction followed by ensemble machine learning classifiers outperformed individual classifiers**, achieving higher accuracy, sensitivity, and specificity.

Challenges in Prediction of ASD

The prediction and diagnosis of ASD present several challenges due to the heterogeneity of the disorder and the complexity of the data involved. ASD is characterized by a wide range of symptoms and severity levels, making it difficult to develop a one-size-fits-all diagnostic tool (Sethi et al., 2024). Additionally, the lack of large, labeled datasets and the variability in data quality across studies pose significant hurdles in training robust models. Many studies rely on small sample sizes, which can lead to overfitting and poor generalization to new data(Zaman et al., 2021) .

Another challenge is the need for interpretable models. While complex models like deep neural networks offer high accuracy, their "black-box" nature makes it difficult for clinicians to understand the decision-making process, which is crucial for gaining their trust and integrating these models into clinical practice (Mohammed et al., 2021). Moreover, existing models often struggle with biases in the training data, which can lead to unfair predictions, particularly across different demographic groups (Arumugam et al., 2021).

Proposed Solution Approach

For this project, the selected approach involves using the ResNet50 deep learning architecture for preliminary ASD detection from facial images, followed by an ensemble machine learning approach for confirmatory diagnosis using the AQ-10 dataset. ResNet50 is chosen for its ability to handle deep networks efficiently through residual connections, which help mitigate the vanishing gradient problem and allow the network to learn more complex features (Narala et al., 2023).

The use of ResNet50 for facial image analysis is supported by several studies that have demonstrated its effectiveness in extracting relevant features from images for classification tasks(Carette et al., 2018; Sun et al., 2023). In this context, ResNet50 will be used to identify potential ASD indicators from facial images, providing an initial diagnosis that can be further validated using traditional machine learning methods.

The confirmatory diagnosis will be conducted using an ensemble of logistic regression, random forest, and XGBoost, with soft voting as the ensemble strategy. This approach leverages the strengths of each model: logistic regression provides a baseline linear model, random forest captures complex interactions between features, and XGBoost offers high performance through gradient boosting (Bose & Seth, 2023; Hemu et al., 2022). Soft voting allows the ensemble to consider the predicted probabilities from each model, leading to a more balanced and accurate final prediction.

This two-step approach—using deep learning for feature extraction and machine learning for final classification—combines the strengths of both methodologies, potentially leading to a more accurate and reliable ASD detection system.

The pursuit of this project is to explore multiple approaches; and create **an easy-to-train, easier-to-deploy** model using ML and DL techniques. The **goal is to enable parents and guardians in identifying autism in children, very early on,** so that appropriate care and support system can be timely and effectively provided to the concerned.
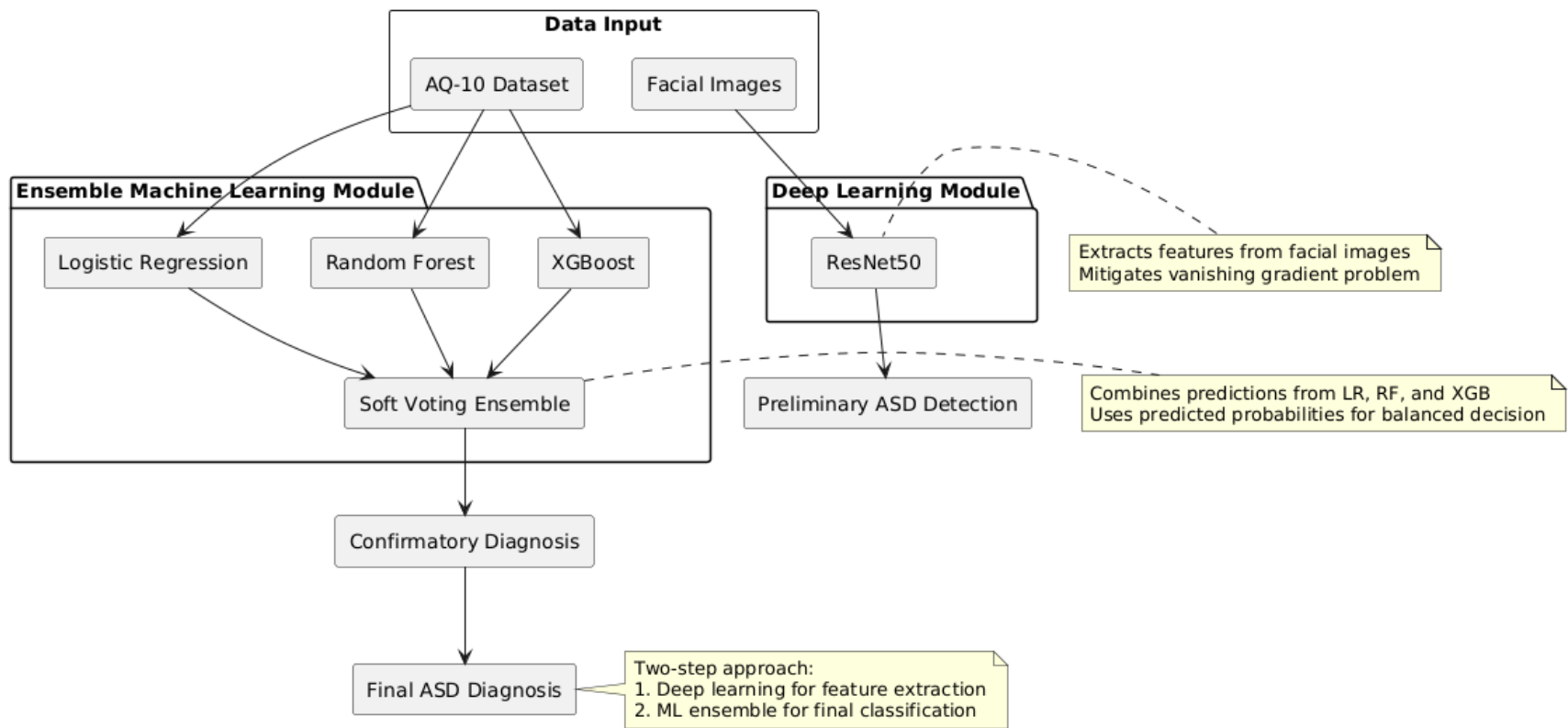
Data Input

AQ-10 Dataset Facial Images

Ensemble Machine Learning Module

Logistic Regression Random Forest XGBoost

Deep Learning Module

ResNet50

Extracts features from facial images
Mitigates vanishing gradient problem

Soft Voting Ensemble

Preliminary ASD Detection

Combines predictions from LR, RF, and XGB
Uses predicted probabilities for balanced decision

Confirmatory Diagnosis

Final ASD Diagnosis

Two-step approach:
1. Deep learning for feature extraction
2. ML ensemble for final classification

Fig 1 Architecture Diagram of the Proposed System

Solution Approach

Deep Learning for Preliminary Diagnosis

Data-set

The dataset required for this segment was taken from (Kaggle, 2022). The dataset contained a total of **2652 images with 1327 images of Autism diagnosed children**. The image dataset was undertaken through multiple pre-processing techniques and then finally split into training and testing data in the 80-20 proportion.

Image-Loading and Preprocessing

Image files were loaded by creating tf.data.Dataset objects to ensure optimized data loading. This was achieved by using a built in function of TensorFlow. This automatically created file labeling based on the directory structure.

```
train_ds_init = image_dataset_from_directory(
    data_dir,
    validation_split=0.2,
    subset="training",
    seed=3,
    image_size=(image_height, image_height),
    batch_size=batch_size
)
```

Further, all image data are set to a fixed image size of 256 X 256 pixels to ensure data consistency. The seed is set to 3 for both training and testing dataset to ensure re-producibility. 80% training data was loaded and augmented while the validation dataset was loaded separately without augmentation.

```
image_height = 256
batch_size=50
```

Image Augmentation

To enhance the model's overall performance and generalizability(Chollet, 2021), techniques of random horizontal and vertical flipping , random rotation, and random zooming (up-to 20%) were implemented using **sequential data augmentation model**.

```
data_augmentation = tf.keras.Sequential([
    # Randomly flip the images horizontally and vertically
    RandomFlip("horizontal_and_vertical"),
    RandomRotation(0.2),  # Randomly rotate images by up to 20%
    RandomZoom(0.2),  # Randomly zoom into the images by up to 20%
])
```

By applying these transformations, the models were exposed to a broader range of data variations. While other suggested approaches include brightness and contrast augmentations, this implementation could not incorporate those approaches into its current scope.

Model Architecture

A sequential model containing a pretrained ResNet50 model was created by excluding the fully connected topmost layers. This transfer learning leveraged features learned on a large dataset (ImageNet) and therefore, enabled accelerated learning.

```python
pretrained_model = ResNet50(include_top=False,
                            input_shape=(256, 256, 3),
                            pooling='avg',
                            weights='imagenet')
```

The internal layers of the pre-trained model were then **frozen to prevent it from further learning** during the training process.

```python
for layer in pretrained_model.layers:
    layer.trainable = False
```

This model evolved **with addition of a few more layers** including a flattening layer, two dense layers with different activation functions, a batch normalization layer and a dropout layer.

```python
resnet_model.add(pretrained_model)
resnet_model.add(Flatten())
resnet_model.add(Dense(512, activation='relu',kernel_regularizer=l2(0.001)))
resnet_model.add(Dense(1, activation='sigmoid'))
resnet_model.add(BatchNormalization())  # Batch Normalization Layer
resnet_model.add(Dropout(0.5))  # Adding Dropout for Regularization
```

The flattening layer **converted the 2D output of ResNet50 layer into 1D vector**. The dense layer with 512 units allowed for learning task-specific features while the second dense layer with single unit and 'sigmoid' activation function enabled classification into 2 categories. Batch normalization was implemented **to ensure a better model stability** while dropout approach of **regularization helped us avoid data-overfitting**(Chollet, 2021).

Model Compilation

The model was compiled using Adam optimizer with a learning rate of 0.001. 'sparse_categorical_crossentropy' loss function was used as loss function.

```
# Compile Model
resnet_model.compile(optimizer=Adam(learning_rate=0.001),
                     loss='sparse_categorical_crossentropy',
                     metrics=['accuracy'])
```

Adam is a popular optimization algorithm in deep learning due to its ability to adjust the learning rate of each parameter individually, making it well-suited for handling sparse gradients and noisy data. **Adam combines the advantages of two other popular optimizers, AdaGrad and RMSProp**, making it highly effective for a wide range of problems(Chollet, 2021).

Early Stopping callback was used to monitor validation loss with a patience of 3 epochs.

```
# Early Stopping Callback
early_stopping = EarlyStopping(monitor='val_loss', patience=3, restore_best_weights=True)
```

Use of early stopping prevented data overfitting(Chollet, 2021). At a reasonable patience value of 3, the model stopped training once it no longer improves, saving time and resources. The model restored the best weights when stopping.

The model was then trained for 6 epochs by providing the respective training and validation data sets.

```
# Train Model with Early Stopping
epochs = 6
history = resnet_model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=epochs,
    callbacks=[early_stopping]  # Include Early Stopping
)
```

Setting smaller number of epochs with early stopping ensured our model did not over train. The trained model was then saved for testing and future usage purposes.

```
resnet_model.save('/content/drive/MyDrive/resnet_model3.h5')
```

Model Prediction

File uploading and reading was a simple and straightforward implementation using 'files.upload()' method. Since this system uploads the image using '*sc2.imread()*', it read images in BGR format by default. The respective conversion was carried out using '*cv2.cv2Color()*' method.

```python
# Get the first (and only) file name
file_name = list(uploaded.keys())[0]

# Load the image using OpenCV
img = cv2.imread(file_name)
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)  # Convert from BGR to RGB
img = cv2.resize(img, target_size)  # Resize the image to the target size
```

The image was expanded along a new axis to create a batch of one image using '*np.expand_dims()*'. This was necessary because most deep learning models expect a batch of images as input.

```python
# Preprocess the image
img_array = np.expand_dims(img, axis=0)  # Expand dims to create batch
img_array = img_array / 255.0  # Normalize the image to [0, 1]
```

The image was normalized by dividing by 255.0, converting pixel values from the range [0, 255] to [0, 1]. The image was then displayed using '*plt.imshow()*' from Matplotlib, with the axes turned off for a cleaner visualization.

```python
# Display the image
plt.imshow(img)
plt.axis('off')  # No axes for the image
plt.show()
```

The uploaded image was then passed into the model using '*model.predict()*' method for prediction.

```python
# Make prediction
predictions = model.predict(img_array)
predicted_class_index = np.argmax(predictions, axis=-1)
predicted_class_name = class_names[predicted_class_index[0]]
```

Machine Learning for Confirmatory Diagnosis

Data-set

The dataset used for this component was the AQ-10 Behavioral dataset with Demographic details. A total of **4 separate data files** from (Kaggle, n.d.-a, n.d.-b, n.d.-c) were used to train the proposed model. These 4 files provided a total of 2947 records. The combination of these 4 files enabled the creation of a diverse database covering children as well as adults.

The combined dataset contained a total of **19 columns with ten columns containing AQ-10** score. The AQ-10 (Autism Spectrum Quotient-10) is a brief screening tool used to identify autistic traits in adults. It's a shortened version of the full Autism Spectrum Quotient (AQ) questionnaire, which contains 50 items. The **AQ-10 was developed to provide a quick and efficient method** for healthcare professionals to assess whether an individual might benefit from a more comprehensive autism evaluation.

Addition to the AQ-10 score, the **dataset contained demographic details** such as age, gender, ethnicity. The dataset contained decent proportion of imbalance among classes which were handled in data preprocessing steps.

Data Pre-processing before Data Merge

To get started, multiple essential libraries were imported in python required to process task at hand. Then 4 separate csv files were loaded into the system.

```python
# Load the four datasets
d1 = pd.read_csv('/content/drive/MyDrive/audata/text/Autism-Child-Data.csv')
d2 = pd.read_csv('/content/drive/MyDrive/audata/text/train.csv')
d3 = pd.read_csv('/content/drive/MyDrive/audata/text/X_prepared.csv')
d4 = pd.read_csv('/content/drive/MyDrive/audata/text/Toddler Autism dataset July 2018.csv')
```

Upon the initial data exploration, discrepancies in data entry in certain column were detected. This was handled by renaming these 'varying' entries using a dictionary.

```python
d1 = d1.rename(columns={
    'A1_Score': 'A1',
    'A2_Score': 'A2',
    'A3_Score': 'A3',
    'A4_Score': 'A4',
    'A5_Score': 'A5',
    'A6_Score': 'A6',
    'A7_Score': 'A7',
    'A8_Score': 'A8',
    'A9_Score': 'A9',
    'A10_Score': 'A10',
    'jundice': 'Jaundice',
    'Class':'Class/ASD',
    'ethnicity': 'Ethnicity',
    'relation': 'Who completed the test'
})
```

```python
d2 = d2.rename(columns={
    'A1_Score': 'A1',
    'A2_Score': 'A2',
    'A3_Score': 'A3',
    'A4_Score': 'A4',
    'A5_Score': 'A5',
    'A6_Score': 'A6',
    'A7_Score': 'A7',
    'A8_Score': 'A8',
    'A9_Score': 'A9',
    'A10_Score': 'A10',
    'jundice': 'Jaundice',
    'Class':'Class/ASD',
    'ethnicity': 'Ethnicity'
})
```

```python
d3 = d3.rename(columns={
    'A1_Score': 'A1',
    'A2_Score': 'A2',
    'A3_Score': 'A3',
    'A4_Score': 'A4',
    'A5_Score': 'A5',
    'A6_Score': 'A6',
    'A7_Score': 'A7',
    'A8_Score': 'A8',
    'A9_Score': 'A9',
    'A10_Score': 'A10',
    'jundice': 'Jaundice',
    'austim': 'Class/ASD',
    'ethnicity': 'Ethnicity'
})
```

This renaming enabled presence of uniform entries across the most crucial data, the clinical measure of behavioral analysis. Further up, on the above 4 csv files, renaming and standardization was conducted for other entries such as '**Sex**'. '**Class/ASD Traits**' and '**Age_Mons**'.

```python
d2 = d2.rename(columns={'jaundice':'Jaundice'})
d3 = d3.rename(columns={'jaundice':'Jaundice'})
d4 = d4.rename(columns={
    'Sex':'gender',
    'Class/ASD Traits ':'Class/ASD',
    'Age_Mons':'age',

})
```

In one of the csv files, the age was given in months, which too was converted into years to ensure general consistency in the data.

```python
d4['age'] = (d4['age'] / 12).round(3)
```

Then the four separate csv files were merged to create on single data set. This dataset now contained **2947 entries, all cleaned and standardized to certain extent**.

Data Preprocessing after Data Merge

After creating one single datafile, un-necessary columns such as '**ID', 'used_app_before', 'Case_No'** etc were dropped to create on new dataset with just the required fields. This enabled us to focus only on relevant features.

```python
dataCB = data.drop(columns=['ID', 'result', 'Case_No','used_app_before','Family_mem_with_ASD','Qchat-10-Score',
                'austim','contry_of_res','age_desc','Who completed the test','relation'])
```

The new dataset formed the following structure:

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	age	gender	Ethnicity	Jaundice	Class/ASD
0	1	1	0	0	1	1	0	1	0	0	6.0	M	Others	no	NO
1	1	1	0	0	1	1	0	1	0	0	6.0	M	Middle Eastern	no	NO
2	1	1	0	0	0	1	1	1	0	0	6.0	M	Others	no	NO
3	0	1	0	0	1	1	0	0	0	1	5.0	F	Others	yes	NO
4	1	1	1	1	1	1	1	1	1	1	5.0	M	Others	yes	YES

The entry values for Gender contain different notations, each of which were standardized to **'M'** and **'F'** notation.

```
replacements = {
    'f': 'F',
    'm': 'M',
    0: 'M',
    1: 'F'
}
dataCB['gender'] = dataCB['gender'].replace(replacements)
```

Handling '**Ethnicity**' entries required a bit of extra attention and care. A significant chunk of entries contained numeric encoding; some had no entry (?), while others contained spaces before (or after) the value. A dictionary was created to clean this column as below:

```
replacements = {
    "'Middle Eastern '": 'Middle Eastern',   # Remove extra spaces and quotes
    "Middle Eastern ": 'Middle Eastern',     # Trim trailing space
    "?": 'Others',                            # Replace unknown entries with 'Others'
    # Convert numeric codes to 'Others'
    10: 'Others',
    8: 'Others',
    2: 'Others',
    1: 'Others',
    5: 'Others',
    0: 'Others',
    11: 'Others',
    4: 'Others',
    9: 'Others',
    6: 'Others',
    3: 'Others',
    7: 'Others'
}
```

Further, all entries under '**Ethnicity**' that contained '**NaN**' were converted into '**Others'.** The idea of filling missing **'Ethnicity'** values with '**Other**' provided a simple and straightforward way to get a clean, consistent data. This study assumed no significant relationship between ethnicity and ASD onset.

```python
dataCB['Ethnicity'] = dataCB['Ethnicity'].replace(replacements)
# Fill any remaining NaNs in 'Ethnicity' with 'Others'
dataCB['Ethnicity'] = dataCB['Ethnicity'].fillna('Others')
```

The entry values of '**others**' were converted to '**Others**'.

```python
replacements = {
    'others': 'Others'
}
dataCB['Ethnicity'] = dataCB['Ethnicity'].replace(replacements)
```

Next up, the entries of '**Jaundice**' column were encoded into numeric code from its original values '**yes**'/'**no**'. This made the **dataset more interpretable**(Aurélien Géron, 2019)**.**

```python
replacements = {
    0: 'no',
    1: 'yes'
}
dataCB['Jaundice'] = dataCB['Jaundice'].replace(replacements)
```

Similarly, '**Class/ASD**' entries were standardized to '**YES**'/ '**NO**' to **ensure consistency and clarity.**

```python
replacements = {
    0: 'NO',
    1: 'YES',
    'No': 'NO',
    'Yes': 'YES'
}
dataCB['Class/ASD'] = dataCB['Class/ASD'].replace(replacements)
```

The '**age**' colum was converted to numeric data type and any conversion errors that came-by were handled by coercing them to '**NaN**'. Then these '**NaN**' values were filled using the media age.

```python
# Convert 'age' column to numeric, coercing errors to NaN
dataCB['age'] = pd.to_numeric(dataCB['age'], errors='coerce')
# Handle NaN values in 'age' by filling with the median age
dataCB['age'].fillna(dataCB['age'].median(), inplace=True)
```

Finally, the '**Class/ASD**' column, which was the target variable, was converted to binary values (1 for '**YES**' and 0 for '**NO**').

```python
# Convert 'Class/ASD' values from 'YES'/'NO' to binary 1/0
dataCB['Class/ASD'] = dataCB['Class/ASD'].map({'YES': 1, 'NO': 0})
```

Next up, the categorical column such as '**gender**', '**Ethnicity**' and '**jaundice**' were encoded using one-hot encoding. This created **binary colums** for each category. The first category was dropped to avoid the dummy variable trap.

```python
categorical_cols = ['gender', 'Ethnicity', 'Jaundice']
dataCB = pd.get_dummies(dataCB, columns=categorical_cols, drop_first=True)
```

Final structure of the dataset was obtained as below:

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	...	Ethnicity_South Asian	Ethnicity_Turkish	Ethnicity_White European	Ethnicity_White-European	Ethnicity_asian	Ethnicity_black	Ethnicity_middle eastern	Ethnicity_mixed	Ethnicity_south asian	Jaundice_yes
0	1	1	0	0	1	1	0	1	0	0	...	False	False	False	False	False	False	False	False	False	False
1	1	1	0	0	1	1	0	1	0	0	...	False	False	False	False	False	False	False	False	False	False
2	1	1	0	0	0	1	1	1	0	0	...	False	False	False	False	False	False	False	False	False	False
3	0	1	0	0	1	1	0	0	0	1	...	False	False	False	False	False	False	False	False	False	True
4	1	1	1	1	1	1	1	1	1	1	...	False	False	False	False	False	False	False	False	False	True

Feature Extraction

The final processed dataset contained a total of 26 columns, out of which only 15 data columns were selected for our model training. The details of the features are as follows:

- **A1-A10 Scores:** These binary scores provided for the most important (and clinically verified) measures to predict Autism Spectrum Disorder as they directly reflected the behavioral patterns of the individual,
- **Age:** Since the expression of Autism traits vary with the age, this feature helped us explore (and possibly understand) the likelihood of autism at different developmental stages of the individual.
- **Gender:** This featured allowed the model to account for potential gender differences in autism diagnosis.
- **Ethnicity:** Ethnicity could influence the diagnosis and presentation of autism due to cultural, genetic and environmental factors. This feature helped in capturing such variations, if any.
- **Jaundice:** The model would attempt to explore any potential connection between jaundice at birth with neurological conditions, including autism.
- **Class/ASD:** This was the target variable for the model.

Model Initialization

The dataset was split into features ('**X**') and target variable ('**Y**').

```python
X = dataCB.drop('Class/ASD', axis=1)
y = dataCB['Class/ASD']
```

Then the feature set was split 70-30 into training and test set.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

The training dataset was then fit into the three ML models selected for this component: Logistic Regression, Random Forest and XGBoost.

```
logreg.fit(X_train, y_train)
rf.fit(X_train, y_train)
xgboost.fit(X_train, y_train)
```

The selection of models attempts to cover a range of **approaches from simple linear models (Logistic Regression) to more complex models** allowing for a robust comparison of each model with the other.

Further an ensemble of these three models was created by implementing 'soft-voting' technique.

```
ensemble = VotingClassifier(estimators=[
    ('logreg', logreg),
    ('rf', rf),
    ('xgboost', xgboost)
], voting='soft')
```

The use of soft voting classifier enabled the system to take advantage of the strength of each model by averaging their probabilistic predictions(Aurélien Géron, 2019). The idea was **to achieve better performance than that of an individual model**. Then this ensemble was trained on the same training data.

```
ensemble.fit(X_train, y_train)
```

Model Outcome and Performance

Resnet-50 for Preliminary Diagnosis

The proposed DL model **performed with 76.54% accuracy** after training for 8 epochs. Due to the timeline constraint of this submission, further fine-tuning of the model had to be shifted for 'future workout'. The detail of model performance after each epoch is as below:

```
Epoch 1/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 766s 17s/step - accuracy: 0.6459 - loss: 1.8228 - val_accuracy: 0.6943 - val_loss: 1.4621
Epoch 2/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 723s 17s/step - accuracy: 0.7085 - loss: 1.3858 - val_accuracy: 0.7415 - val_loss: 1.2047
Epoch 3/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 757s 17s/step - accuracy: 0.7114 - loss: 1.2072 - val_accuracy: 0.7075 - val_loss: 1.2061
Epoch 4/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 707s 16s/step - accuracy: 0.7260 - loss: 1.1124 - val_accuracy: 0.7283 - val_loss: 1.0797
Epoch 5/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 745s 17s/step - accuracy: 0.7537 - loss: 1.0307 - val_accuracy: 0.7302 - val_loss: 1.0283
Epoch 6/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 720s 17s/step - accuracy: 0.7411 - loss: 0.9699 - val_accuracy: 0.7358 - val_loss: 0.9618
Epoch 7/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 737s 17s/step - accuracy: 0.7538 - loss: 0.9134 - val_accuracy: 0.7396 - val_loss: 0.9069
Epoch 8/8
43/43 ━━━━━━━━━━━━━━━━━━━━ 733s 17s/step - accuracy: 0.7654 - loss: 0.8562 - val_accuracy: 0.7264 - val_loss: 0.8857
```

Over the course of training, a steady increase in training accuracy was observed, rising from 0.6459 in the first epoch to 0.7654 in the final epoch. This upward trend indicated that the model effectively learned to predict the correct classes within the training set, while a corresponding decrease in training loss from 1.8228 to 0.8562 demonstrated that the model's errors on the training data were consistently minimized. Despite this improvement, it was noted that the **pace of accuracy gain slowed towards the later epochs**, suggesting that the **model was nearing convergence** but might still have had the potential to improve **further with additional training or increased model complexity**.
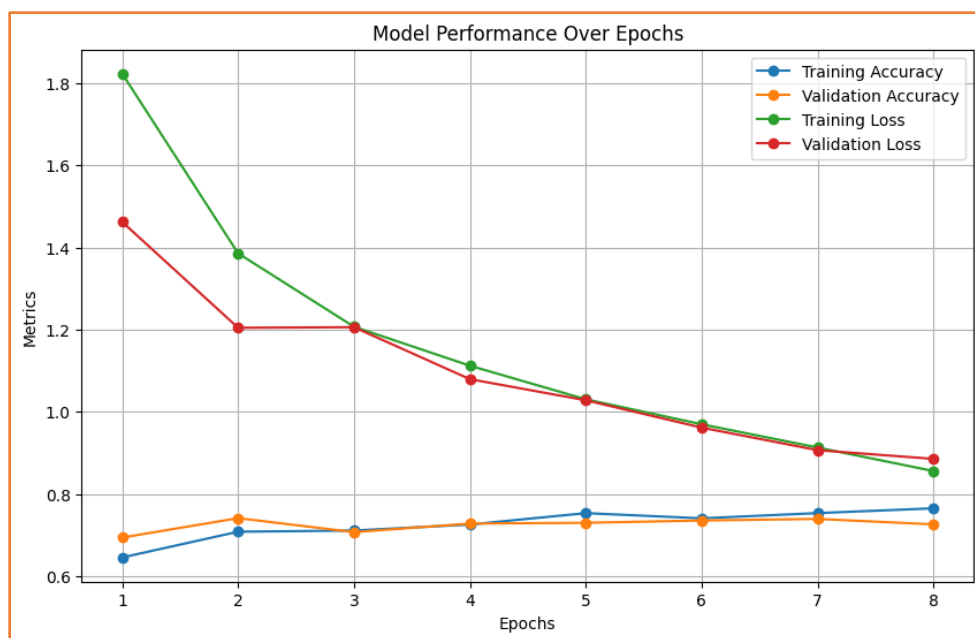

Fig 2 Learning Curve for the Proposed Model

In terms of validation performance, an initial improvement in accuracy was noted, peaking at 0.7415 during the second epoch, after which the accuracy fluctuated slightly but generally stabilized around 0.73. The validation loss showed an overall decrease from 1.4621 in the first epoch to 0.8857 in the final epoch, reflecting a gradual enhancement in the model's generalization capabilities. However, **a slight divergence between the training and validation accuracy was observed, particularly towards the later epochs, hinting at the**

onset of overfitting. The close proximity of the final validation loss to the training loss **suggested that overfitting was minimal**, though it remained a concern to be monitored in subsequent training.

Further, the model was analyzed based on its ROC plot, which showed that the model performed better than random guessing, as indicated by the curve rising above the diagonal baseline. The AUC was measured at 0.79, suggesting that the model had a good ability to distinguish between positive and negative classes.
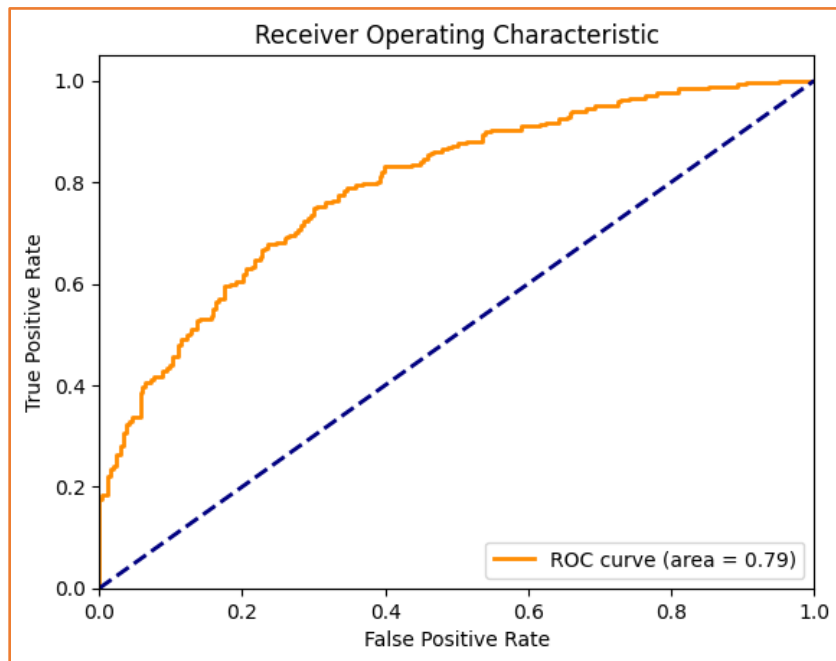

Fig 3 ROC plot of the Proposed Model Training

The curve displayed a steady increase in the true positive rate with relatively few false positives, indicating effective prediction of positive instances. However, some concavity in the curve suggested that there were areas where the balance between true positives and false positives could have been optimized further. Overall, while the model demonstrated an appreciable level of discriminative ability, there was still room for improvement to achieve better

Ensemble ML for Confirmatory Diagnosis

The performance of the ensemble model, comprising Logistic Regression, Random Forest, and XGBoost, was evaluated using accuracy, precision, recall, and F1-score metrics. Each individual model demonstrated strong performance, with the Logistic Regression model achieving an overall accuracy of 0.88, characterized by a precision of 0.92 for class 0 and 0.83 for class 1. The Random Forest model performed slightly better, with an accuracy of 0.90 and a balanced precision of 0.91 for class 0 and 0.88 for class 1. XGBoost also showed

a robust performance with a similar accuracy of 0.90, where class 0 had a precision of 0.91, and class 1 had a precision of 0.88.

The ensemble model, which combined the predictions of all three models, achieved the highest accuracy of 0.92. This improved performance was reflected in the increased precision and recall for both classes, with class 0 having a precision of 0.94 and class 1 a precision of 0.89. The F1-score for the ensemble model also improved, reaching 0.93 for class 0 and 0.89 for class 1, demonstrating that the ensemble approach effectively leveraged the strengths of the individual models to produce more accurate and reliable predictions across both classes.
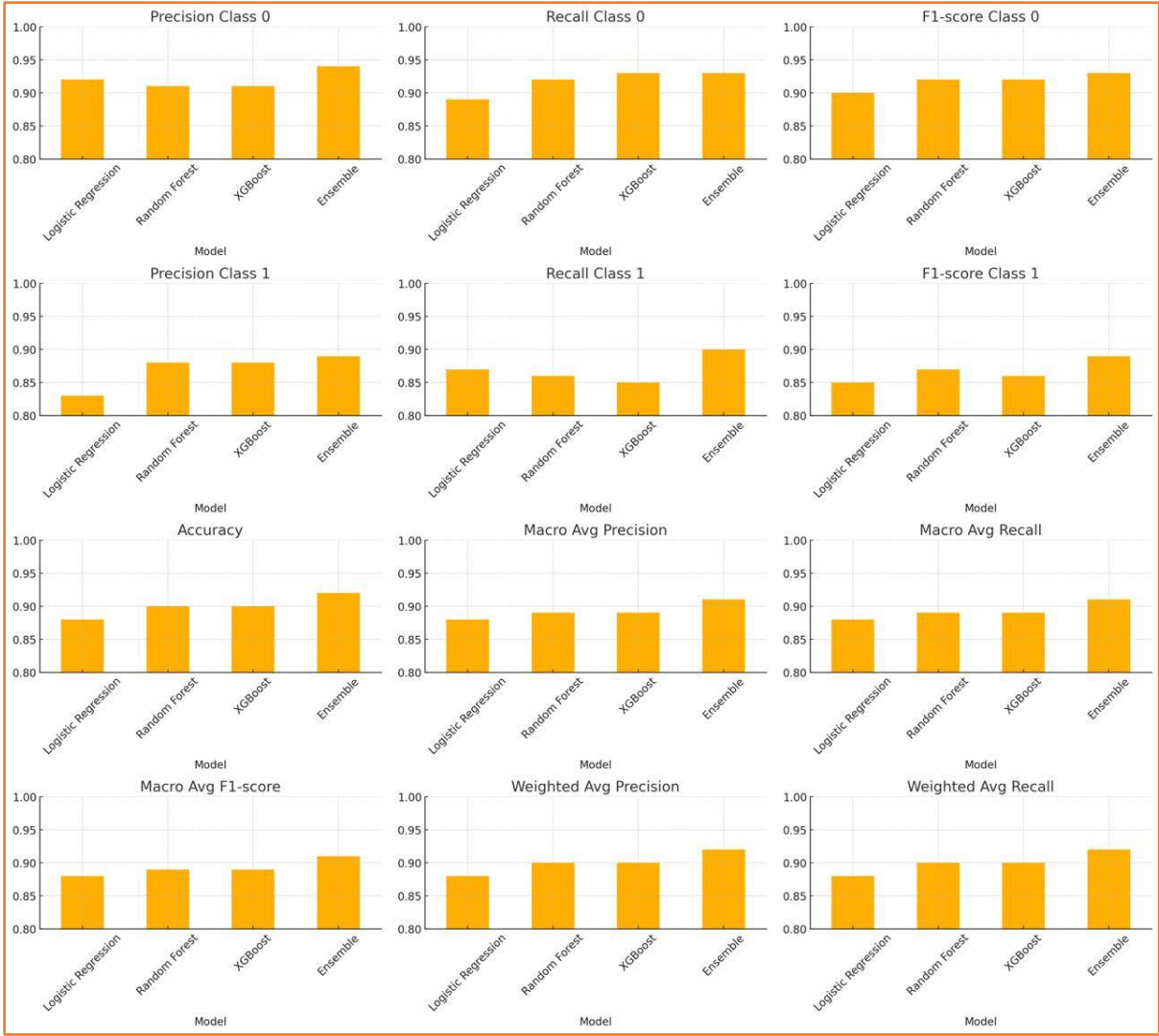
Fig 4 Performance of Proposed ML Model

The AUC (Area Under the Curve) values for all models are high, with Logistic Regression achieving an **AUC of 0.94, Random Forest 0.95**, and both XGBoost and the Ensemble model achieving an **AUC of 0.96**. The close clustering of the curves near the top-left corner of the

plot indicates **that all models have a high true positive rate with a relatively low false positive** rate. The Ensemble model and XGBoost slightly outperform the others, as indicated by their marginally higher AUC values, suggesting **a slight advantage in predictive accuracy**. Overall, the models performed well, with the Ensemble model demonstrating the best overall performance**, slightly edging out XGBoost and the other algorithms**.
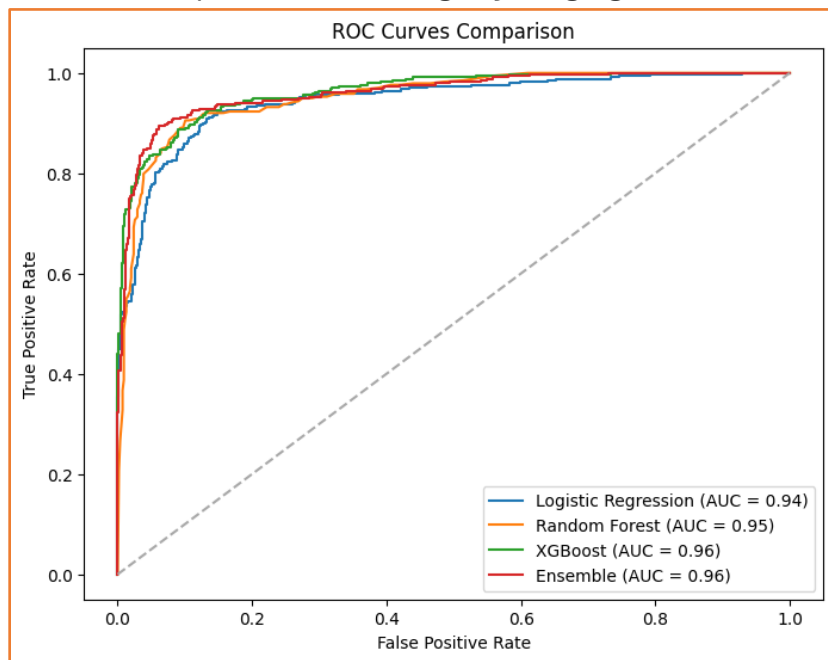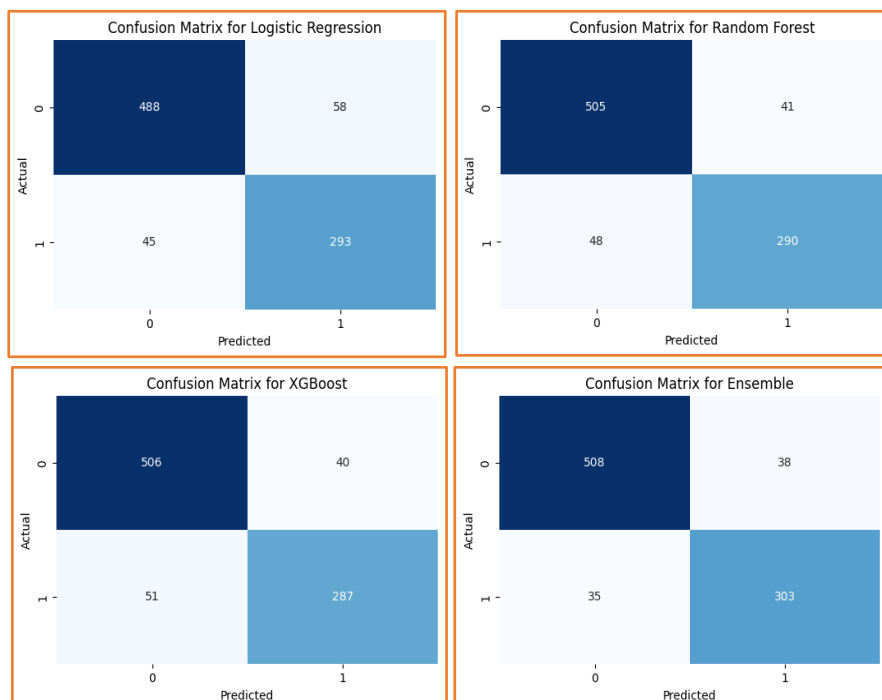

Fig 5 ROC Curve of the Proposed Model

Fig 6 Confusion Matrix of the Proposed Model
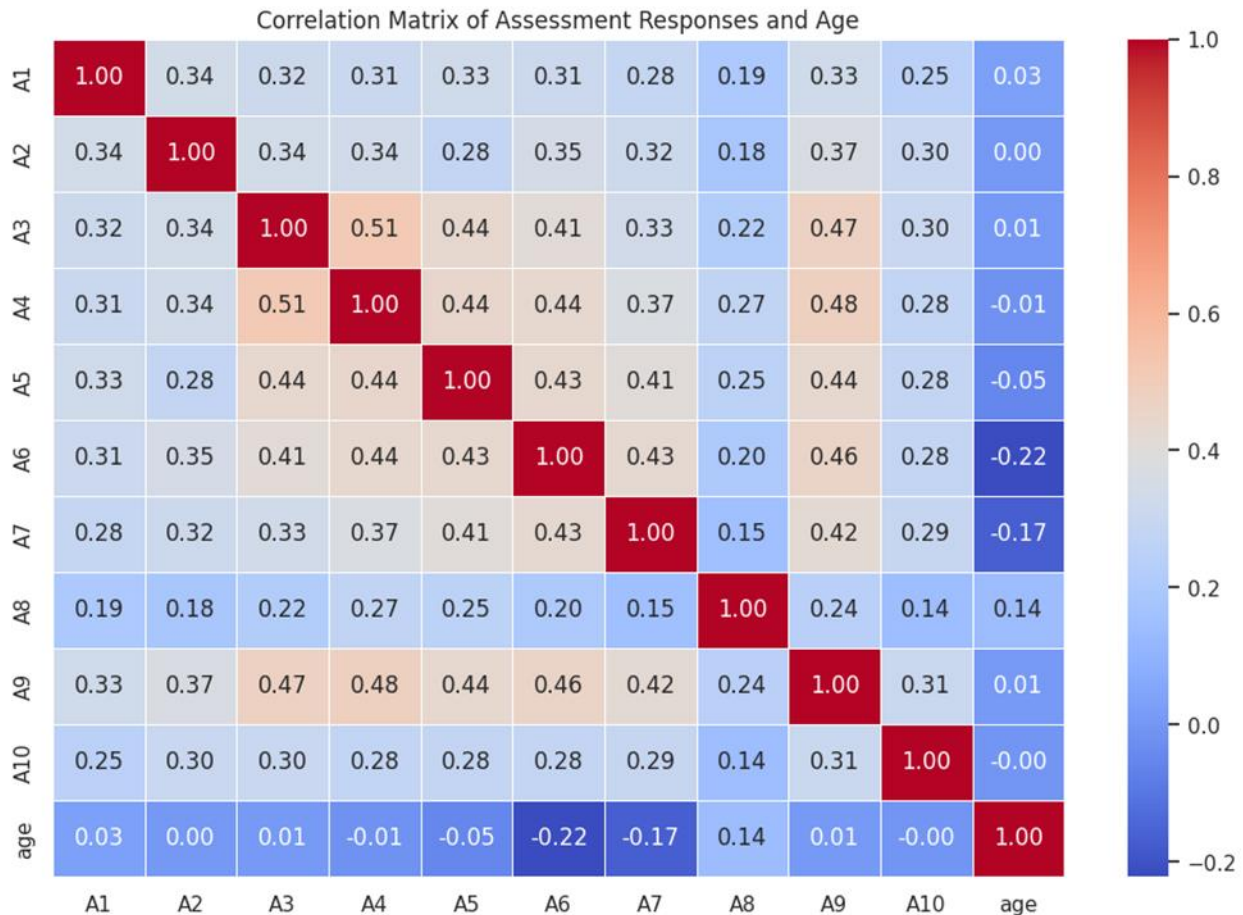
Additional Insights:

Fig 7 Correlation Martix of Assesment Responses and Age

The heatmap shown above was used to illustrate the correlation matrix for the assessment responses (A1 to A10) and age. It was observed that most of the assessment responses exhibited low to moderate correlations with each other, indicating some level of interdependence among these features. This interdependence could have been explored further in predictive modeling or deeper analysis. Additionally, it was found that age had very low correlations with the assessment responses, suggesting that the responses were relatively independent of the individuals' age in the dataset. This correlation analysis was useful in understanding the relationships between different variables, which could have been crucial for tasks such as feature selection in machine learning or identifying key factors in assessments.

Future Scope

The future scope of this project lies in several critical areas that could enhance its impact and applicability. Firstly, further fine-tuning of the deep learning model ResNet50, could lead to improved accuracy in preliminary ASD detection from facial images. Given that the current model achieved a respectable accuracy of 76.54% after 8 epochs, additional epochs, hyperparameter tuning, and the inclusion of more advanced data augmentation techniques could potentially elevate the model's performance.

Moreover, integrating other deep learning architectures or exploring ensemble deep learning approaches might provide a broader perspective on feature extraction and classification accuracy, particularly in handling the diversity and complexity of ASD presentations.

In the context of the confirmatory diagnosis phase, expanding the dataset to include more diverse demographic information and behavioral data could significantly improve the robustness and generalizability of the ensemble machine learning model.

Exploring alternative ensemble strategies, such as stacking with more complex meta-learners, could further refine the balance between precision, recall, and F1 scores across different classes.

Additionally, enhancing the interpretability of these models through explainable AI techniques would be critical for gaining trust and adoption in clinical settings. Addressing these aspects would position this project to make a substantial contribution to early ASD detection and diagnosis, particularly in low-resource settings where access to specialized diagnostic tools is limited.

Ethical Concerns

The proposed system and report raise several ethical concerns that need careful consideration. One primary concern is the potential for bias in the machine learning models, particularly given the demographic imbalances and variability in data quality across different groups, which could lead to unfair or inaccurate predictions. This is especially critical in the context of ASD diagnosis, where misclassification could have serious implications for individuals and their families.

Further, the use of facial recognition and neuroimaging data involves sensitive personal information, raising issues around privacy, consent, and data security. Ensuring that these systems are transparent, explainable, and developed with input from diverse stakeholders, including those from underrepresented groups, is crucial to prevent harm and ensure that the benefits of such technology are equitably distributed.

Conclusion

In conclusion, this project explored the development of a comprehensive system for the early detection and confirmatory diagnosis of Autism Spectrum Disorder (ASD) using a combination of deep learning and machine learning techniques. The ResNet50 model demonstrated promising results in preliminary ASD detection through facial image analysis, while the ensemble machine learning approach, incorporating Logistic Regression, Random Forest, and XGBoost, showed enhanced accuracy and reliability in confirmatory diagnosis using behavioral and demographic data.

The proposed system, while innovative, requires careful implementation and continuous evaluation to ensure it serves its intended purpose without inadvertently causing harm. As such, future work will focus on refining the models, expanding the datasets, and enhancing the interpretability of the system to ensure its effectiveness and ethical integrity in real-world applications. This project, we hope, will contribute to the growing body of research aimed at improving early ASD detection and diagnosis and it was a fantastic journey pursuing this idea (and developing the prototype!).

Project Repository

Project files, dataset and all the necessary components can be accessed at this link:

https://github.com/karki-dennis/ASD-Prediction-Using-AI-ML

https://github.com/Acharya-jyu/ASD-Prediction-Using-AI-ML

References

Akter, T., Shahriare Satu, M., Khan, M. I., Ali, M. H., Uddin, S., Lio, P., Quinn, J. M. W., & Moni, M. A. (2019). Machine Learning-Based Models for Early Stage Detection of Autism Spectrum Disorders. IEEE Access, 7. https://doi.org/10.1109/ACCESS.2019.2952609

Allely, C. S. (2022). What Is Autism Spectrum Disorder (ASD)? In Autism Spectrum Disorder in the Criminal Justice System. https://doi.org/10.4324/9781003212195-1

APA. (2019). Diagnostic and statistical manual of mental disorders (DSM-5®): American Psychiatric Pub; 2013. J. Physiother. Res., Salvador, 9(2).

Arumugam, S. R., Balakrishna, R., Khilar, R., Manoj, O., & Shylaja, C. S. (2021). Prediction of Autism Spectrum Disorder in Children using Face Recognition. Proceedings - 2nd International Conference on Smart Electronics and Communication, ICOSEC 2021, 1246–1250. https://doi.org/10.1109/ICOSEC51865.2021.9591679

Aurélien Géron. (2019). Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems. In O'Reilly Media.

Autism Speaks. (2017). AUTISM AND HEALTH: A SPECIAL REPORT BY AUTISM SPEAKS Advances in Understanding and Treating the Health Conditions that Frequently Accompany Autism. https://www.autismspeaks.org/science-news/autism-and-health-special-report-autism-speaks

Bose, S., & Seth, P. (2023). Screening of Autism Spectrum Disorder using Machine Learning Approach in Accordance with DSM-5. 2023 7th International Conference on Electronics, Materials Engineering and Nano-Technology, IEMENTech 2023. https://doi.org/10.1109/IEMENTech60402.2023.10423494

Carette, R., Elbattah, M., Dequen, G., Guerin, J. L., & Cilia, F. (2018). Visualization of eye-tracking patterns in autism spectrum disorder: Method and dataset. 2018 13th International Conference on Digital Information Management, ICDIM 2018. https://doi.org/10.1109/ICDIM.2018.8846967

Chollet, F. (2021). Deep Learning with Python, Second Edition. In Deep Learning with Python.

Elsabbagh, M., Divan, G., Koh, Y. J., Kim, Y. S., Kauchali, S., Marcín, C., Montiel-Nava, C., Patel, V., Paula, C. S., Wang, C., Yasamy, M. T., & Fombonne, E. (2012). Global Prevalence of Autism and Other Pervasive Developmental Disorders. Autism Research, 5(3). https://doi.org/10.1002/aur.239

Hemu, A. A., Mim, R. B., Ali, M. M., Nayer, M., Ahmed, K., & Bui, F. M. (2022). Identification of Significant Risk Factors and Impact for ASD Prediction among Children Using Machine Learning Approach. 2022 2nd International Conference on Advances in Electrical,

Computing, Communication and Sustainable Technologies, ICAECT 2022. https://doi.org/10.1109/ICAECT54875.2022.9808043

Jaffar, S. S., & Abdulbaqi, H. A. (2022). Facial Expression Recognition in Static Images for Autism Children using CNN Approaches. Proceedings - CSCTIT 2022: 5th College of Science International Conference on Recent Trends in Information Technology, 202–207. https://doi.org/10.1109/CSCTIT56299.2022.10145689

Kaggle. (n.d.-a). AQ-10 Autism Dataset. Retrieved August 13, 2024, from https://www.kaggle.com/datasets/fabdelja/autism-screening-for-toddlers/data

Kaggle. (n.d.-b). AQ10 Dataset Autism. Retrieved August 13, 2024, from https://www.kaggle.com/code/vishwasmishra1234/final-with-99-accuracy/

Kaggle. (n.d.-c). Autism AQ-10 Dataset. Retrieved August 13, 2024, from https://www.kaggle.com/code/ashishpawar511/autism-screening-with-neural-network/

Kaggle. (2022). Facial Image Dataset.

Khosla, Y., Ramachandra, P., & Chaitra, N. (2021). Detection of autistic individuals using facial images and deep learning. CSITSS 2021 - 2021 5th International Conference on Computational Systems and Information Technology for Sustainable Solutions, Proceedings. https://doi.org/10.1109/CSITSS54238.2021.9683205

Lever, A. G., & Geurts, H. M. (2016). Psychiatric Co-occurring Symptoms and Disorders in Young, Middle-Aged, and Older Adults with Autism Spectrum Disorder. Journal of Autism and Developmental Disorders, 46(6). https://doi.org/10.1007/s10803-016-2722-8

Li, X., Dvornek, N. C., Zhuang, J., Ventola, P., & Duncan, J. S. (2018). Brain biomarker interpretation in ASD using deep learning and fMRI. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11072 LNCS. https://doi.org/10.1007/978-3-030-00931-1_24

Mohammed, M. B., Salsabil, L., Shahriar, M., Tanaaz, S. S., & Fahmin, A. (2021). Identification of Autism Spectrum Disorder through Feature Selection-based Machine Learning. 24th International Conference on Computer and Information Technology, ICCIT 2021. https://doi.org/10.1109/ICCIT54785.2021.9689805

Naik, S. K. R., Deepa, M., Ruhi, P. B., Prakash, S., & Royal, U. J. (2023). Determination and Diagnosis of Autism Spectrum Disorder using Efficient Machine Learning Algorithm. 2023 3rd International Conference on Intelligent Technologies, CONIT 2023. https://doi.org/10.1109/CONIT59222.2023.10205718

Narala, M. S. V. S. K., Vemuri, S., & Kattula, C. (2023). A Comparative Study for Autism spectrum disorder Using EfficientNet, ResNet50. 2023 International Conference on Computational Intelligence, Networks and Security, ICCINS 2023. https://doi.org/10.1109/ICCINS58907.2023.10450124

National Institute of Mental Health (NIMH). (2024). Autism Spectrum Disorder. https://www.nimh.nih.gov/health/topics/autism-spectrum-disorders-asd

Oma, K. S., Mondal, P., Khan, N. S., Rizvi, M. R. K., & Islam, M. N. (2019). A Machine Learning Approach to Predict Autism Spectrum Disorder. 2nd International Conference on Electrical, Computer and Communication Engineering, ECCE 2019. https://doi.org/10.1109/ECACE.2019.8679454

Sadiq, S., Castellanos, M., Moffitt, J., Shyu, M. L., Perry, L., & Messinger, D. (2019). Deep learning based multimedia data mining for autism spectrum disorder (ASD) diagnosis. IEEE International Conference on Data Mining Workshops, ICDMW, 2019-November. https://doi.org/10.1109/ICDMW.2019.00124

Sethi, A., Khan, K., Katarya, R., & Yingthawornsuk, T. (2024). Empirical Evaluation of Machine Learning Techniques for Autism Spectrum Disorder. Proceeding - 12th International Electrical Engineering Congress: Smart Factory and Intelligent Technology for Tomorrow, IEECON 2024. https://doi.org/10.1109/iEECON60677.2024.10537970

Sun, C., Sun, D., & Liu, Z. P. (2023). Detecting Autism Spectrum Disorder from Functional Magnetic Resonance Imaging Data with an Ensemble Machine Learning Method. Proceedings - 2023 5th International Conference on Intelligent Medicine and Image Processing, IMIP 2023, 13–20. https://doi.org/10.1109/IMIP57114.2023.00010

Thabtah, F., & Peebles, D. (2020). A new machine learning model based on induction of rules for autism detection. Health Informatics Journal, 26(1). https://doi.org/10.1177/1460458218824711

Venkata Sai Krishna Narala, M. S., Vemuri, S., & Kattula, C. (2023). Prediction of Autism Spectrum Disorder Using Efficient Net. 2023 9th International Conference on Advanced Computing and Communication Systems, ICACCS 2023, 1139–1143. https://doi.org/10.1109/ICACCS57279.2023.10112807

WHO. (2013). Autism spectrum disorders & other developmental disorders From raising awareness to building capacity.

Zaman, N., Ferdus, J., & Sattar, A. (2021). Autism Spectrum Disorder Detection Using Machinelearning Approach. 2021 12th International Conference on Computing Communication and Networking Technologies, ICCCNT 2021. https://doi.org/10.1109/ICCCNT51525.2021.9579522

Zeidan, J., Fombonne, E., Scorah, J., Ibrahim, A., Durkin, M. S., Saxena, S., Yusuf, A., Shih, A., & Elsabbagh, M. (2022). Global prevalence of autism: A systematic review update. In Autism Research (Vol. 15, Issue 5). https://doi.org/10.1002/aur.2696