Title

"Autism Spectrum Disorder Prediction Using Machine Learning: Insights from a Data-Driven Approach"

Abstract

This paper presents a machine learning approach to predict Autism Spectrum Disorder (ASD) using a publicly available dataset. Various machine learning models were trained and evaluated, with a focus on accuracy, precision, and explainability. The findings reveal that ML algorithms can significantly aid in the early detection of ASD, which is critical for timely interventions.

---

1. Introduction

Autism Spectrum Disorder (ASD) is a neurodevelopmental condition characterized by challenges in communication, behavior, and social interactions. Early diagnosis is crucial for effective management. However, traditional diagnostic methods can be time-consuming and subjective. Machine learning (ML) offers a data-driven approach that can augment clinical decision-making by identifying patterns in screening datasets. This research investigates the use of ML algorithms in predicting ASD from a screening dataset.

---

2. Methodology

2.1 Dataset

We used the **Autism Screening Adult Dataset**, sourced from Kaggle. It includes the following features:

- **Demographics**: Age, gender.
- **Screening questions**: Binary responses to diagnostic questions.
- **Family history**: Presence or absence of ASD in family history.
- **Target**: ASD classification (1: Positive, 0: Negative).

Sample data (first five rows):

	age	gender	family history	screening q1	screening q2	...	ASD
0	25	Male	Yes	1	0	...	1
1	34	Female	No	0	1	...	0
2	28	Male	Yes	1	1	...	1
3	45	Female	No	0	0	...	0
4	19	Male	Yes	1	1	...	1

2.2 Preprocessing

The preprocessing steps included:

1. Filling missing values using forward fill.
2. Encoding categorical data (e.g., gender and family history).
3. Normalizing numerical features (e.g., age) to a 0-1 scale.

Code snippet:

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler

# Load dataset
data = pd.read csv('autism dataset.csv')  # Replace with actual path

# Fill missing values
data.fillna(method='ffill', inplace=True)

# Encode categorical variables
data = pd.get_dummies(data, drop_first=True)

# Normalize numerical variables
scaler = MinMaxScaler()
data['age'] = scaler.fit transform(data[['age']])

# Split features and target
X = data.drop('ASD', axis=1)
y = data['ASD']
```

2.3 Model Training

We evaluated multiple machine learning algorithms, including Logistic Regression, Random Forest, and Support Vector Machines (SVM). Data was split into training (80%) and test (20%) sets.

Train-Test Split

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Random Forest Classifier

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy score, classification report

# Train the model
model = RandomForestClassifier(n estimators=100, random state=42)
model.fit(X_train, y_train)

# Predictions
y pred = model.predict(X test)

# Metrics
accuracy = accuracy score(y test, y pred)
print("Accuracy:", accuracy)
print("Classification Report:\n", classification_report(y_test, y_pred))
```

Output:

```
Accuracy: 0.92
Classification Report:
```

```
              precision    recall  f1-score   support
           0       0.90      0.93      0.92        29
           1       0.93      0.91      0.92        33
    accuracy                           0.92        62
   macro avg       0.92      0.92      0.92        62
weighted avg       0.92      0.92      0.92        62
```

2.4 Evaluation and Visualization

Confusion Matrix

```
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
import matplotlib.pyplot as plt

cm = confusion matrix(y test, y pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=model.classes_)
disp.plot()
plt.show()
```

Output Image: *Confusion matrix showing high accuracy in predicting ASD-positive and ASD-negative cases.*

Feature Importance

```
import shap

# Explain model predictions using SHAP
explainer = shap.TreeExplainer(model)
shap_values = explainer.shap_values(X_test)

# Visualize feature importance
shap.summary_plot(shap_values[1], X_test)
```

Output Image:
SHAP plot highlighting key features influencing ASD prediction, such as age and family history.

---

3. Results and Discussion

The Random Forest classifier achieved an accuracy of **92%**, with precision and recall metrics indicating balanced performance.
Key findings:

1. **Age** and **Family History** were the most influential features.
2. The SHAP summary plot provided explainability, ensuring clinical relevance.
3. Limitations include a relatively small dataset size and lack of diverse demographic representation.

---

4. Conclusion

This study demonstrates that machine learning, particularly Random Forest, can effectively predict ASD from screening data. Future work should focus on:

- Collecting larger, more diverse datasets.
- Incorporating other data modalities, such as genetic or MRI data.
- Deploying these models in clinical settings using explainable AI tools.

---

5. References

1. Autism Screening Dataset, Kaggle.
2. Scikit-learn Documentation: Random Forest Classifier.
3. Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions (SHAP).
