YUNUS Mujahid Olalekan

EEG/2018/096

Date: Sunday, March 17, 2024.

TASK COMPLETION REPORT

Original Task Assigned

Develop a high-performing model for EEG-based diagnosis of Schizophrenia

Detailed Description of Accomplishments

1. **Data Ingest**: I was able to extract the EEG data contained in the edf files provided by Mr Emmanuel Olateju and store them in easily accessible csv format. Likewise, I was able to extract the patients' details from the gnr files into a separate dataframe. Finally, I merged the two dataframes into a single dataframe which forms the dataset that I used for the modelling.
2. **Exploratory Data Analysis:** After creating the required dataset, I explored it to gain insights into the data, dropped unnecessary features and converted some categorical features to numerical features for smooth modeling.
3. **Cross-validation and Validation of Machine Learning models**: I attempted to look for the best model for the created dataset by cross-validating and validating different machine learning models. Support Vector Classifier gave the highest accuracy of 68.16% among all the Machine Learning models. This low accuracy warranted the trial of neural network models. The results are shown below:

```
Cross Validation accuracies for the LogisticRegression(max_iter=1000) = [0.6122449  0.73469388 0.32653061 0.81632653 0.65306122]
Acccuracy score of the  LogisticRegression(max_iter=1000) = 62.86 %
---------------------------------------------------------------
Cross Validation accuracies for the SVC(kernel='linear') = [0.6122449  0.59183673 0.40816327 0.7755102  0.63265306]
Acccuracy score of the  SVC(kernel='linear') = 60.41 %
---------------------------------------------------------------
Cross Validation accuracies for the KNeighborsClassifier() = [0.48979592 0.75510204 0.3877551  0.44897959 0.28571429]
Acccuracy score of the  KNeighborsClassifier() = 47.35 %
---------------------------------------------------------------
Cross Validation accuracies for the RandomForestClassifier(random_state=0) = [0.51020408 0.7755102  0.46938776 0.3877551  0.42857143]
Acccuracy score of the  RandomForestClassifier(random_state=0) = 51.43 %
---------------------------------------------------------------
```

Figure 1: Cross Validation 1

```
Cross Validation accuracies for the GaussianNB() = [0.32653061 0.75510204 0.51020408 0.67346939 0.69387755]
Acccuracy score of the  GaussianNB() = 59.18 %
---------------------------------------------------------------
Cross Validation accuracies for the GradientBoostingClassifier() = [0.51020408 0.83673469 0.57142857 0.42857143 0.44897959]
Acccuracy score of the  GradientBoostingClassifier() = 55.92 %
---------------------------------------------------------------
```

Figure 2: Cross Validation 2

	model used	highest score	best hyperparameters
0	LogisticRegression(max_iter=10000)	0.628571	{'C': 1}
1	SVC()	0.681633	{'C': 1, 'kernel': 'sigmoid'}
2	KNeighborsClassifier()	0.546939	{'n_neighbors': 10}
3	RandomForestClassifier(random_state=0)	0.514286	{'n_estimators': 100}

Figure 3: Validation

4. **Neural Network Modeling:** Upon the first trial of a simple neural network model, it gave an accuracy of 75.6%, which is significantly greater than SVC's highest accuracy of 68.16%. After several architecture optimizations, I achieved the best model with 91.8% accuracy.
5. **Results and Evaluation:** I used accuracy and confusion matrix metrics to evaluate the model. I got the best-performing model by training a simple neural network with standardized time-series EEG data. The accuracy of this best model is 91.84%. The confusion matrix gives 25 true positives, 20 true negatives, 1 false positive, and 3 false negatives.

Subtasks Accomplished

- Data Preparation: Execute necessary processes to make the acquired dataset model-ready (**Completed)**
- Exploratory Data Analysis: Gain insights into the data to know the set of appropriate models (**Completed)**
- Validation of different Models: Test different models without real training to choose a best-performing model (**Completed)**
- Model Optimization: Optimize the best-performing model's architecture and hyperparameters to get higher accuracy. (**Completed)**

Next Steps

- Further Evaluation: Devise more robust processes to evaluate the model
- Documentation**:** Create a comprehensive report detailing the processes involved in the project and the various results obtained.