Assignment 1: Machine Learning Model
Maderi de Meyer
40977676
ITRI 616
Prediction

Alzheimer’s Disease Prediction App

Contents

Introduction

The aim of this project is to predict the likelihood of Alzheimer’s Disease by taking a variety of different aspects into account. It is therefore a classification model that makes use of binary output. Using Python, a machine learning model was built that allows the user to select one of three models to use in training and to evaluate the accuracy of the different methods.

Background

A progressive neurodegenerative disease, called Alzheimer's Disease, can result in devastating symptoms and standard of living. The outcomes can be reduced or even prevented when detected and diagnosed early on (Leifer, 2003). That is what this model aims to achieve by providing the user, preferably a doctor, to enter patient information into the web application, where the model will predict the diagnosis (Alzheimer’s or not).

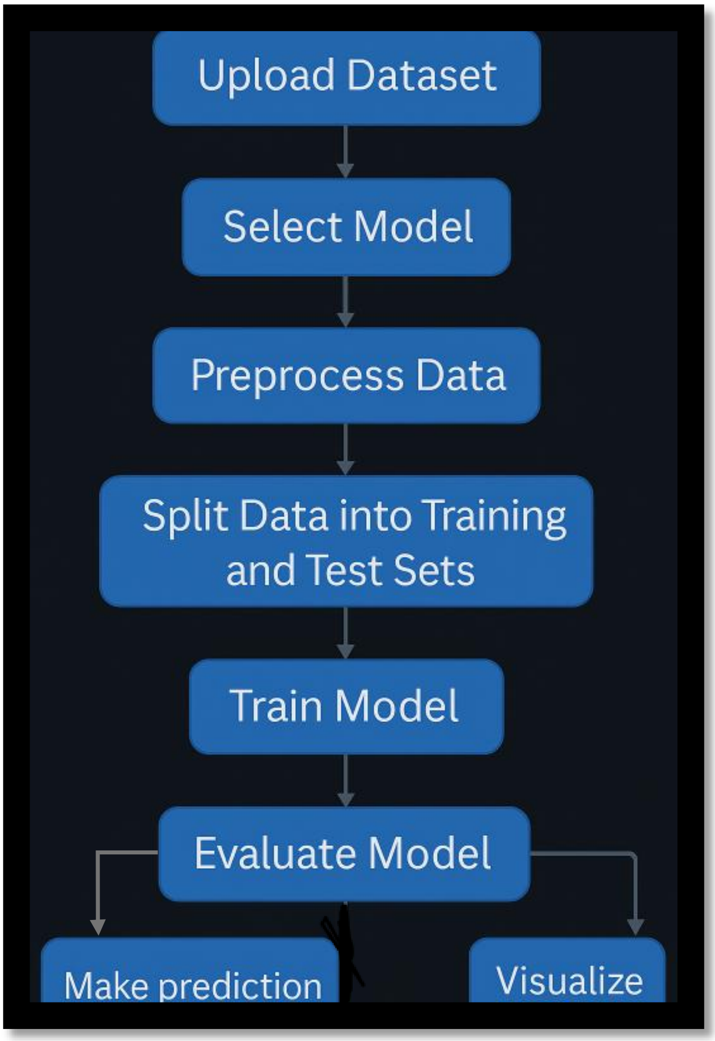
Materials and Methodology

Materials

The data used to train and test the model is all simulated and exported from www.Kaggle.com . The dataset contains 2149 patients’ detailed medical and lifestyle information. Each patient has a diagnosis status, with the value 1 if they are likely to have Alzheimer's Disease, and 0 if not. The numerical and categorical variables consist of:

- Demographics: Age, Gender, Ethnicity and Education Level.
- Lifestyle Factors: BMI, Smoking, Alcohol consumption and Physical activity.
- Medical History: Diabetes, Hypertension, Depression and Head injury.
- Clinical Measurements: Systolic/Diastolic Blood pressure and Cholesterol levels.
- Cognitive Assessments: MMSE, Functional score and Memory complaints.
- Symptoms: Confusion, Disorientation, Forgetfulness, etc.
- Target Variable: Diagnosis (Binary Classification).

Method



The data was pre-processed before training, applying the following steps:

- Dropped non-feature columns, such as DoctorInCharge and PatientID, which is not required for predicting diagnosis.
- Diagnosis was used for the target variable, which results in 0 if Alzheimer's Disease is not likely and 1 if otherwise.
- All the numerical features in the dataset were standardized using StandardScaler to implement uniform scaling across all features.
- Using train_test_split(), the dataset was split into eighty percent for training and twenty percent for the purpose of testing.

The user can select the model that will be used to train the model. The accuracy of each model is displayed, to allow users to compare the different classifiers and ultimately choose the best-performing model.

- Random Forest Classifier is good with feature importance and was given the parameter: random_state=42.
- Logistic Regression is also an option if the user prefers a linear baseline model. It was provided with the parameters: max_iter=2000 and random_state=42.
- Support Vector Machine is a kernel-based classification that was given parameters: probability=True and random_state=42.

The trained models were trained using a standardized training set and the predictions were evaluated using the test data. The model was evaluated using the following metrics:

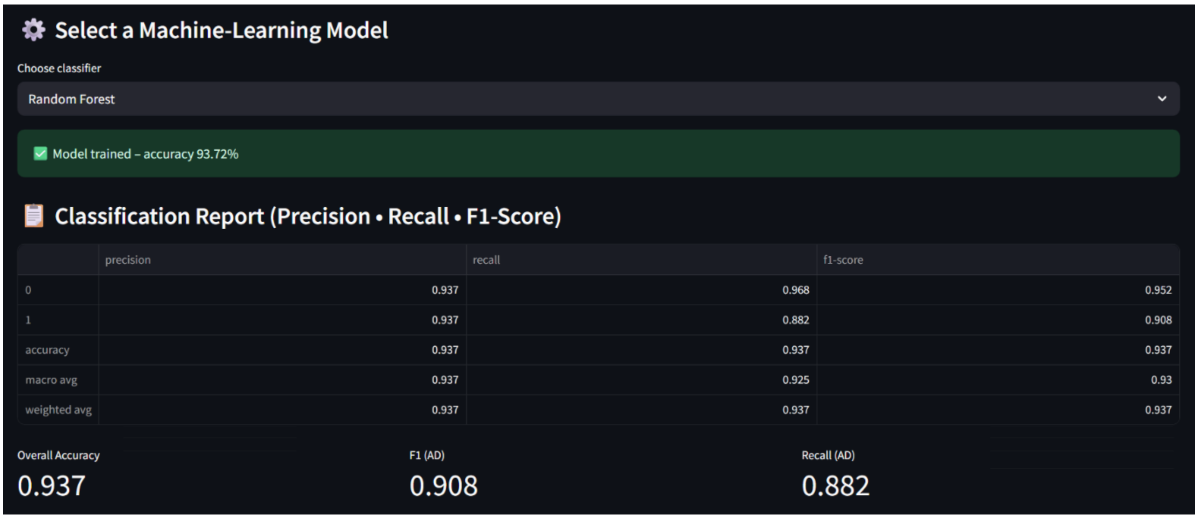
- Accuracy, which is the number of correct predictions overall.
- Precision, providing the proportion of true positives out of all the predicted positives.
- Recall, which is the proportion of actual positives that was predicted correctly.
- F1-Score, representing the mean of precision and recall.
- ROC Curve, providing the user with a visual performance across the thresholds.
- AUC Score, which is the area under the ROC curve. The higher this score, the better the discrimination ability.

To develop this model, Python 3.11 was used, along with streamlit for building the interactive web dashboard. Libraries, such as pandas and numpy, was implemented for data manipulation, scikit-learn for model training, evaluation and preprocessing and seaborn and matplotlib for data and evaluation visualization.

Results and Evaluation

Random Forest Classifier Model

For the Random Forest Classifier, the model performance table is as follows:



Select a Machine-Learning Model

Choose classifier

Random Forest

☑ Model trained – accuracy 93.72%

Classification Report (Precision • Recall • F1-Score)

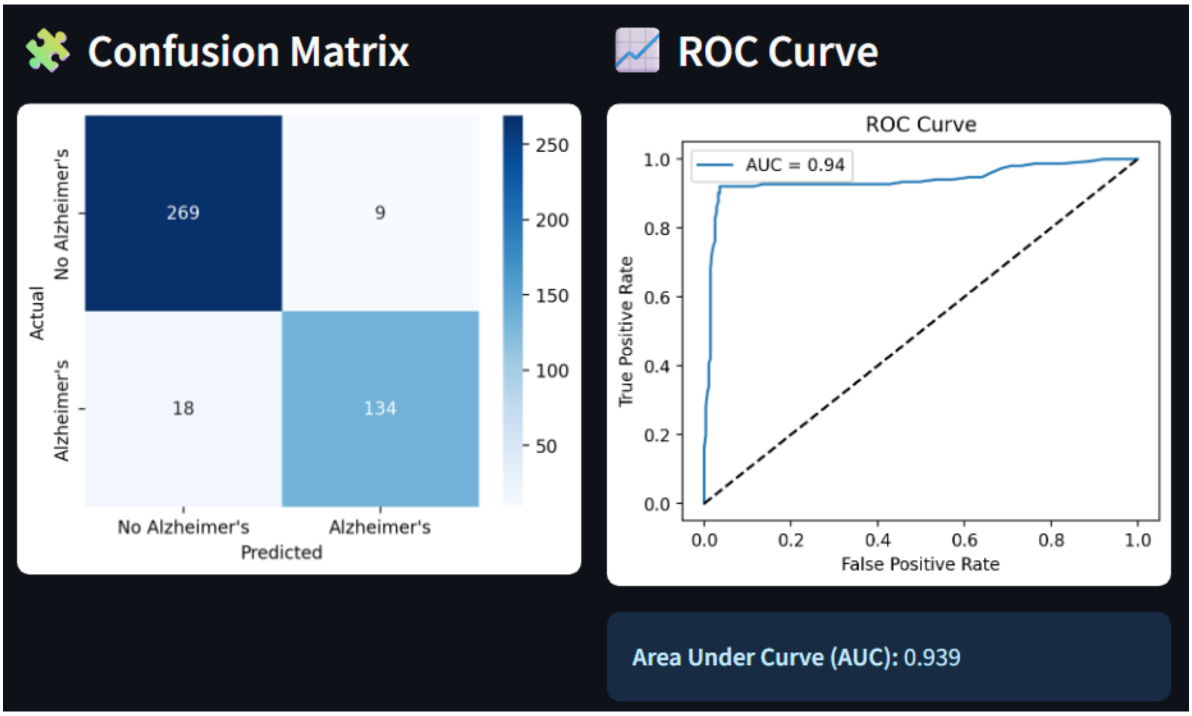
	precision	recall	f1-score
0	0.937	0.968	0.952
1	0.937	0.882	0.908
accuracy	0.937	0.937	0.937
macro avg	0.937	0.925	0.93
weighted avg	0.937	0.937	0.937

Overall Accuracy
0.937

F1 (AD)
0.908

Recall (AD)
0.882

As displayed, the model has an accuracy of 93.72%. Its F1 value is 0.908 and has a Recall value of 0.882. According to the confusion matrix, when the actual diagnosis was 'No Alzheimer's' it predicted 269 of the diagnosis correctly and when the actual diagnosis was 'Alzheimer's', it predicted 134 correctly. This means it is not only accurate, but also sensitive to Alzheimer 's detection. According to the ROC curve this model has a very strong performance level as the blue line (AUC) is sharp. An AUC of 94% indicates that there is a 94% chance that the model will correctly classify a randomly chosen patient.



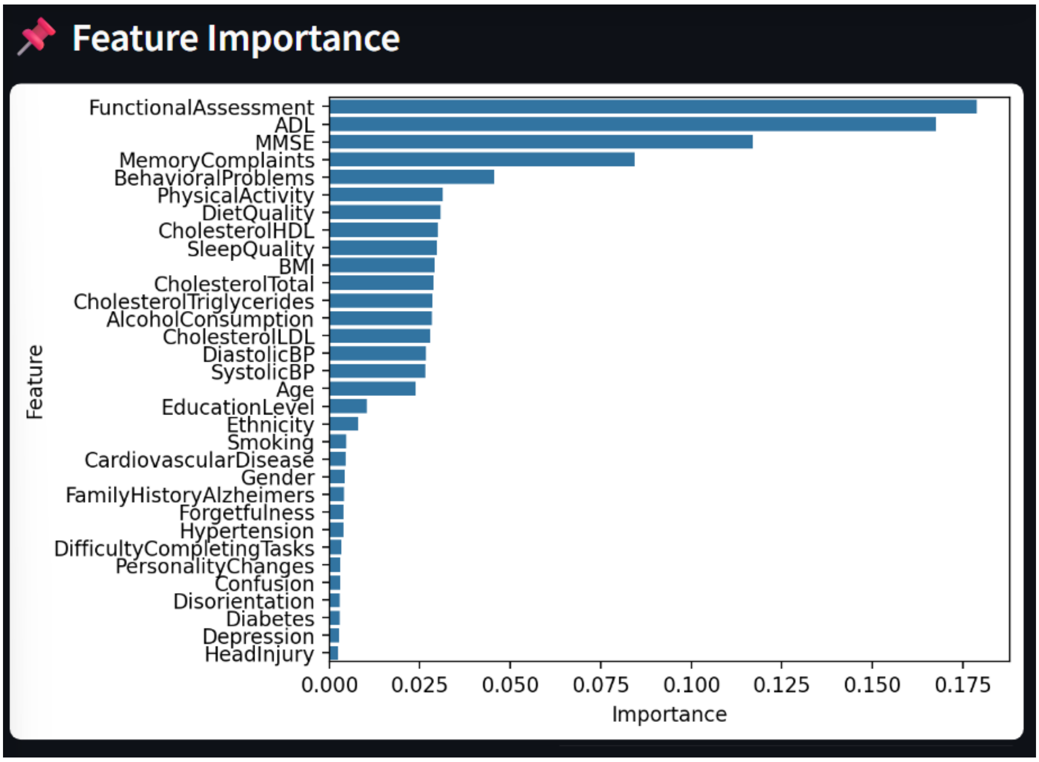
🧩 Confusion Matrix

Actual \ Predicted	No Alzheimer's	Alzheimer's
No Alzheimer's	269	9
Alzheimer's	18	134

📈 ROC Curve

Area Under Curve (AUC): 0.939

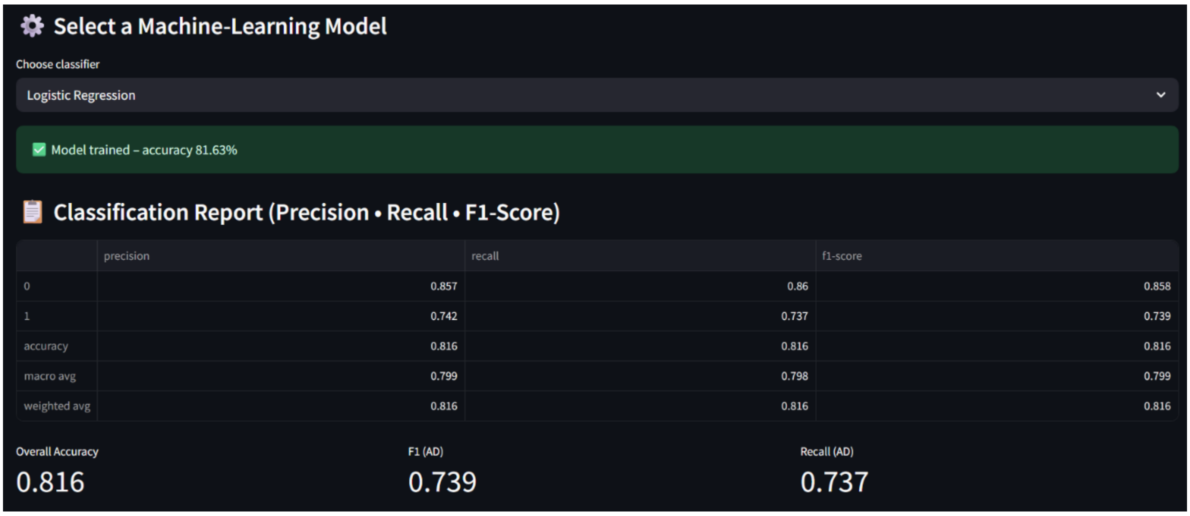
The importance of all the different features in a random forest classifier model is applicable and therefore presented.



📌 Feature Importance

Logistic Regression Model

For the Logistic Regression model, it displayed the following performance table:



⚙ Select a Machine-Learning Model

Choose classifier

Logistic Regression

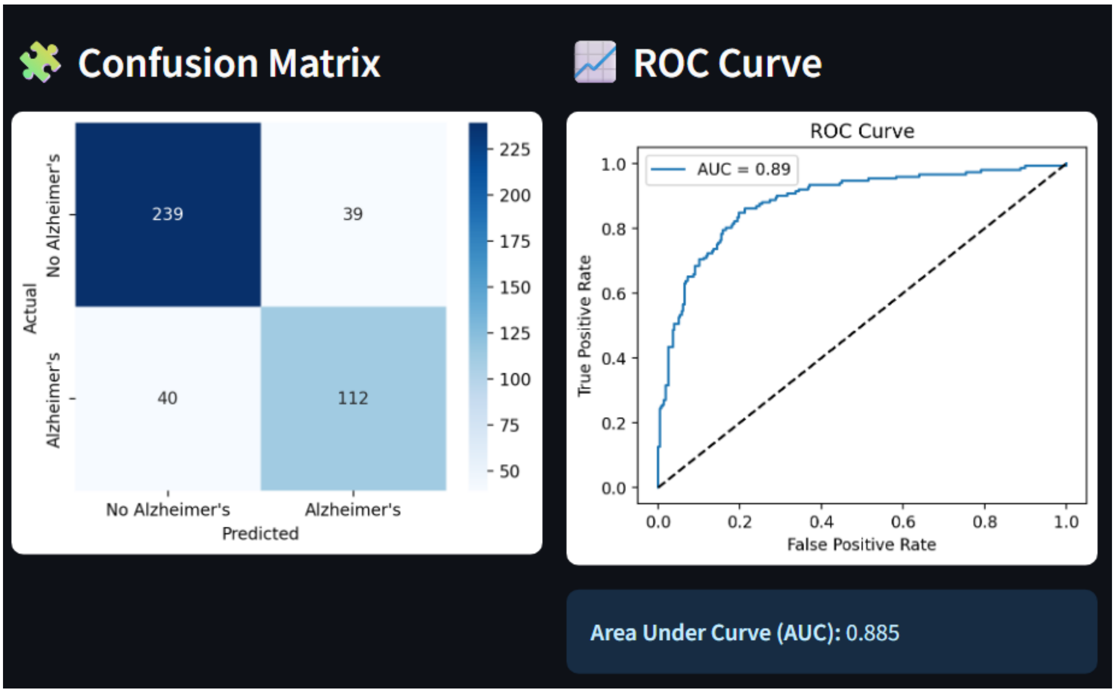
☑ Model trained – accuracy 81.63%

Classification Report (Precision • Recall • F1-Score)

	precision	recall	f1-score
0	0.857	0.86	0.858
1	0.742	0.737	0.739
accuracy	0.816	0.816	0.816
macro avg	0.799	0.798	0.799
weighted avg	0.816	0.816	0.816

Overall Accuracy	F1 (AD)	Recall (AD)
0.816	0.739	0.737

The results show an accuracy of 81.63%, with a 0.739 F1 value and 0.737 recall value. The confusion matrix table indicates that 239 patients was correctly diagnosed by the model when their actual diagnosis was ‘No Alzheimer’s’. 112 patients were correctly diagnosed when their actual diagnosis was ‘Alzheimer’s’. The AUC value indicates that the model can distinguish Alzheimer’s diagnosis from non-Alzheimer’s cases 88.5% of the time.



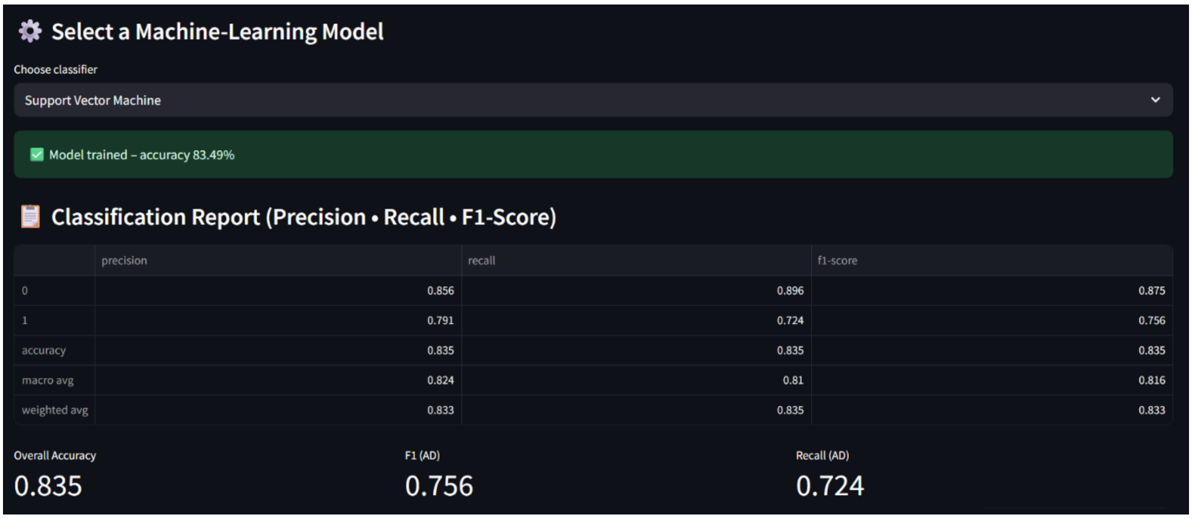
Confusion Matrix

ROC Curve

Area Under Curve (AUC): 0.885

Support Vector Model

Lastly, the support vector machine presented the following classification report:



Select a Machine-Learning Model

Choose classifier

Support Vector Machine

☑ Model trained – accuracy 83.49%

Classification Report (Precision • Recall • F1-Score)

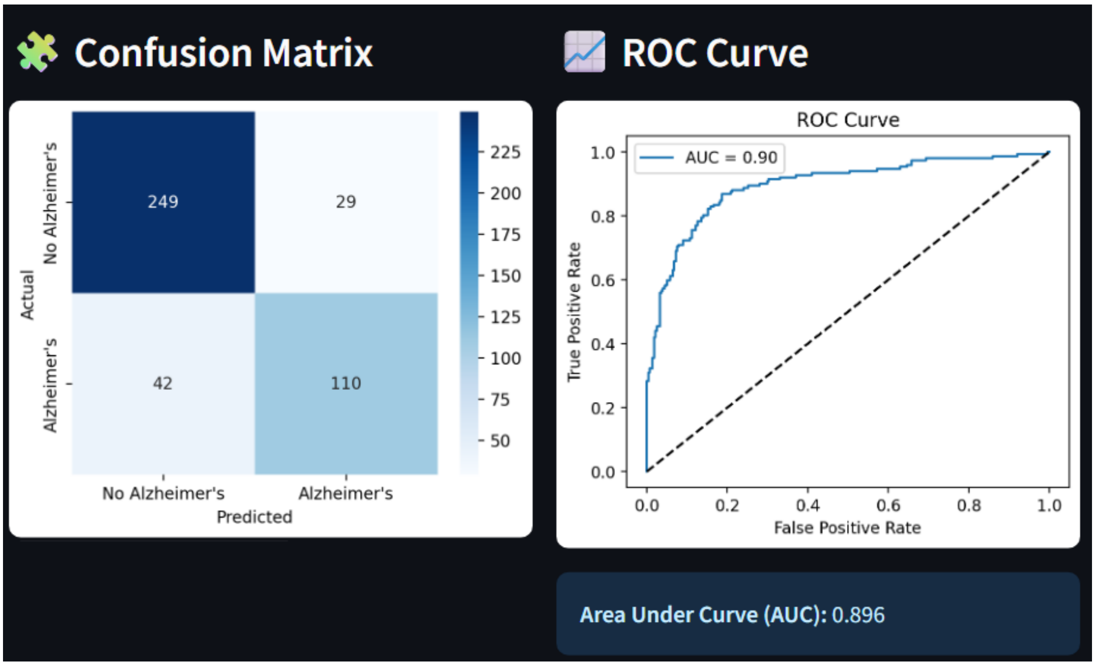
	precision	recall	f1-score
0	0.856	0.896	0.875
1	0.791	0.724	0.756
accuracy	0.835	0.835	0.835
macro avg	0.824	0.81	0.816
weighted avg	0.833	0.835	0.833

Overall Accuracy 0.835

F1 (AD) 0.756

Recall (AD) 0.724

According to this table, this model has an accuracy of 83.49%. It has an F1 value of 0.756 and 0.724 as its Recall value. According to this confusion matrix, it correctly diagnosed 249 'No Alzheimer's' cases and 110 'Alzheimer's' cases. The ROC curve and AUC displays similar results as the logistic regression model and therefore successfully diagnose patients 90% of the time.



Confusion Matrix

ROC Curve

Area Under Curve (AUC): 0.896

Conclusion

The Random Forest classification model was proven most successful for this specific dataset as it has an accuracy of 93.72%. It also has an F1 of 0.908, where the other models have values of only 0.739 and 0.756. The AUC for the Random Forest model is closest to 1.0, with a value of 0.939. The other models have a lower accuracy, AUC, F1 and Recall values, highlighting the most applicable model to be the random forest model.

How to run the web application

1. Download the zip folder named ‘AI Individual Project’.
2. Extract the folder
3. Open Visual Studio Code
4. Select ‘Open Folder ‘
5. Select the folder ‘AI Individual Project’
6. In the terminal of the root folder of the project run:

```
python –version
```

7. If Python is not found, install it from https://www.python.org/downloads/
8. Finally run these commands:

```
pip install -r requirements.txt
streamlit run app.py
```

How to use the Model

1. Upload the .csv file that is located in the extracted folder named ‘AI Individual Project’.
2. Select the model (Random Forest, Logistic Regression and Support Vector Machine)
3. Scroll down to view the Classification Report, Confusion Matrix and ROC Curve graph, Feature Importance (If applicable), Explore Variables section and enter information for a new patient to predict diagnosis.

References

Leifer, B.P., 2003. Early diagnosis of Alzheimer's disease: clinical and economic benefits. *Journal of the American Geriatrics Society*, *51*(5s2), pp. S281-S288.