ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH

HỌC MÁY(CO3117)

Bài tập lớn

Triển khai các mô hình học máy

Giảng viên hướng dẫn: Võ Thanh Hùng
Sinh viên tham gia: 2211212 - Nguyễn Gia Huy
2211180 - Huỳnh Quốc Huy
2211157 - Đàm Đức Huy
2213732 - Cao Nguyễn Văn Trường

Thành phố Hồ Chí Minh, 4/2025

MSSV	Họ và tên	Phân công
2211212	Nguyễn Gia Huy	Tiền xử lí dữ liệu, Huấn luyện mô hình, viết báo cáo
2211157	Đàm Đức Huy	Phân tích dữ liệu, Huấn luyện mô hình, viết báo cáo
2211180	Huỳnh Quốc Huy	Tiền xử lí dữ liệu, Huấn luyện mô hình, thiết kế slide
2213732	Cao Nguyễn Văn Trường	Phân tích dữ liệu, Huấn luyện mô hình, thiết kế slide

Bảng 1: Bảng phân công công việc

MỤC LỤC

1 Giới thiệu bộ dữ liệu **3**

2 Phân tích dữ liệu **6**
2.1 Khám phá dữ liệu 6
2.2 Phân tích mối quan hệ giữa các đặc trưng 11

3 Tiền xử lí dữ liệu **15**
3.1 Biến liên tục 15
3.2 Biến Rời rạc (không có ý nghĩa thứ tự) 15
3.3 Chia dữ liệu huấn luyện và kiểm thử 16

4 Các mô hình dự đoán **17**

5 Kết quả và so sánh các mô hình **19**
5.1 Mô hình Cây quyết định (Decision Tree) 19
5.2 Mô hình K-Nearest Neighbors (KNN) 19
5.3 Mô hình Hồi quy Logistic (Logistic Regression) 19
5.4 Mô hình Support vector machine (SVM) 19
5.5 Tóm tắt kết quả và so sánh 20

6 Code và Bộ dữ liệu **21**

Tham khảo **22**

1 Giới thiệu bộ dữ liệu

Alzheimers disease data là một bộ dữ liệu bao gồm thông tin y tế, lâm sàng và đánh giá chức năng tâm thần của 2.149 bệnh nhân. Mục đích thu thập là để nghiên cứu các yếu tố nguy cơ, hình thành nên bệnh Alzheimer. Từ đó, xây dựng các mô hình học máy để dự đoán liệu bệnh nhân có mắc bệnh Alzheimer hay không.

Bộ dữ liệu gồm 2149 dòng và 35 cột tương ứng với 2149 bệnh nhân và 35 đặc trưng của bệnh nhân. Dưới đây là 35 đặc trưng:

STT	Tên cột	Ý nghĩa	Kiểu dữ liệu
1	PatientID	Mã định danh duy nhất cho mỗi bệnh nhân (4751–6900)	Integer
2	Age	Tuổi bệnh nhân (60–90)	Integer
3	Gender	Giới tính (0=Nam; 1=Nữ)	Binary
4	Ethnicity	Dân tộc (0=Caucasian; 1=African American; 2=Asian; 3=Other)	Categorical
5	EducationLevel	Trình độ học vấn (0=None; 1=High School; 2=Bachelor's; 3=Higher)	Categorical
6	BMI	Body Mass Index (15–40)	Float
7	Smoking	Hút thuốc (0=Không; 1=Có)	Binary
8	AlcoholConsumption	Lượng rượu tiêu thụ hàng tuần (0–20 đơn vị)	Integer
9	PhysicalActivity	Giờ hoạt động thể chất mỗi tuần (0–10)	Float
10	DietQuality	Điểm chất lượng chế độ ăn (0–10)	Integer
11	SleepQuality	Điểm chất lượng giấc ngủ (4–10)	Integer
12	FamilyHistoryAlzheimers	Tiền sử gia đình mắc Alzheimer (0=Không; 1=Có)	Binary
13	CardiovascularDisease	Bệnh tim mạch (0=Không; 1=Có)	Binary
14	Diabetes	Tiểu đường (0=Không; 1=Có)	Binary
15	Depression	Trầm cảm (0=Không; 1=Có)	Binary
16	HeadInjury	Chấn thương sọ não (0=Không; 1=Có)	Binary
17	Hypertension	Tăng huyết áp (0=Không; 1=Có)	Binary
18	SystolicBP	Huyết áp tâm thu (90–180 mmHg)	Integer
19	DiastolicBP	Huyết áp tâm trương (60–120 mmHg)	Integer
20	CholesterolTotal	Cholesterol toàn phần (150–300 mg/dL)	Integer
21	CholesterolLDL	LDL cholesterol (50–200 mg/dL)	Integer
22	CholesterolHDL	HDL cholesterol (20–100 mg/dL)	Integer
23	CholesterolTriglycerides	Triglycerides (50–400 mg/dL)	Integer
24	MMSE	Điểm Mini-Mental State Examination (0–30)	Integer
25	FunctionalAssessment	Điểm đánh giá chức năng (0–10)	Integer
26	MemoryComplaints	Than phiền về trí nhớ (0=Không; 1=Có)	Binary
27	BehavioralProblems	Vấn đề hành vi (0=Không; 1=Có)	Binary
28	ADL	Activities of Daily Living (0–10)	Integer
29	Confusion	Lú lẫn (0=Không; 1=Có)	Binary
30	Disorientation	Mất phương hướng (0=Không; 1=Có)	Binary

STT	Tên cột	Ý nghĩa	Kiểu dữ liệu
31	PersonalityChanges	Thay đổi nhân cách (0=Không; 1=Có)	Binary
32	DifficultyCompletingTasks	Khó hoàn thành công việc (0=Không; 1=Có)	Binary
33	Forgetfulness	Hay quên (0=Không; 1=Có)	Binary
34	Diagnosis	Chẩn đoán Alzheimer (0=Không; 1=Có)	Binary (target)
35	DoctorInCharge	Thông tin bác sĩ phụ trách (giá trị "XXXConfid" cho mọi bản ghi; bảo mật)	Text

Mô hình sẽ dùng các đặc trưng khác để dự đoán biến mục tiêu Diagnosis.

Import dữ liệu:

```
df = pd.read_csv('/kaggle/input/alzheimers-disease-dataset/
    alzheimers_disease_data.csv')
df.head().T
```

Cấu trúc của dữ liệu:

STT	0	1	2	3	4
PatientID	4751	4752	4753	4754	4755
Age	73	89	73	74	89
Gender	0	0	0	1	0
Ethnicity	0	0	3	0	0
EducationLevel	2	0	1	1	0
BMI	22.927749	26.827681	17.795882	33.800817	20.716974
Smoking	0	0	0	1	0
AlcoholConsumption	13.297218	4.542524	19.555085	12.209266	18.454356
PhysicalActivity	6.327112	7.619885	7.844988	8.428001	6.310461
DietQuality	1.347214	0.518767	1.826335	7.435604	0.795498
SleepQuality	9.025679	7.151293	9.673574	8.392554	5.597238
FamilyHistoryAlzheimers	0	0	1	0	0
CardiovascularDisease	0	0	0	0	0
Diabetes	1	0	0	0	0
Depression	1	0	0	0	0
HeadInjury	0	0	0	0	0
Hypertension	0	0	0	0	0
SystolicBP	142	115	99	118	94
DiastolicBP	72	64	116	115	117
CholesterolTotal	242.36684	231.162595	284.181858	159.58224	237.602184
CholesterolLDL	56.150897	193.407996	153.322762	65.366637	92.8697
CholesterolHDL	33.682563	79.028477	69.772292	68.457491	56.874305
CholesterolTriglycerides	162.189143	294.630909	83.638324	277.577358	291.19878
MMSE	21.463532	20.613267	7.356249	13.991127	13.517609
FunctionalAssessment	6.518877	7.118696	5.895077	8.965106	6.045039
MemoryComplaints	0	0	0	0	0
BehavioralProblems	0	0	0	1	0
ADL	1.725883	2.592424	7.119548	6.481226	0.014691

STT	0	1	2	3	4
Confusion	0	0	0	0	0
Disorientation	0	0	1	0	0
PersonalityChanges	0	0	0	0	1
DifficultyCompletingTasks	1	0	1	0	1
Forgetfulness	0	1	0	0	0
Diagnosis	0	0	0	0	0
DoctorInCharge	XXXConfid	XXXConfid	XXXConfid	XXXConfid	XXXConfid

Bộ dữ liệu không chứa giá trị null và mọi hàng đều khác biệt.

2 Phân tích dữ liệu

2.1 Khám phá dữ liệu

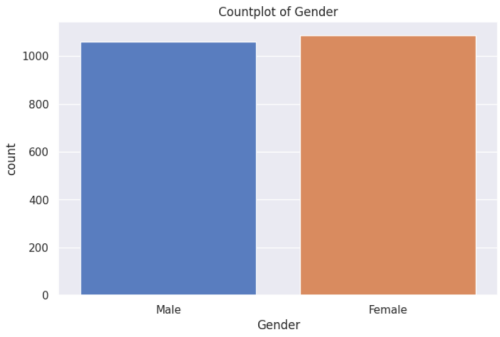
Trước khi tiến hành phân tích, mô hình sẽ loại bỏ hai cột `PatientID` (mã số bệnh nhân) và `DoctorInCharge` (bác sĩ phụ trách) vì chúng không có ý nghĩa trong việc dự đoán.

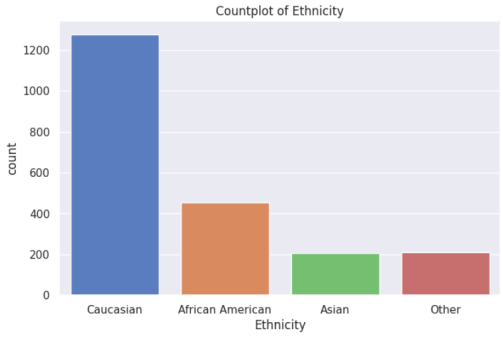
Sau khi loại bỏ, tập dữ liệu còn lại gồm 33 đặc trưng, trong đó có biến mục tiêu `Diagnosis`, và tổng cộng 2149 mẫu. Tập dữ liệu sau đó được tách làm hai phần: dữ liệu rời rạc và dữ liệu liên tục.

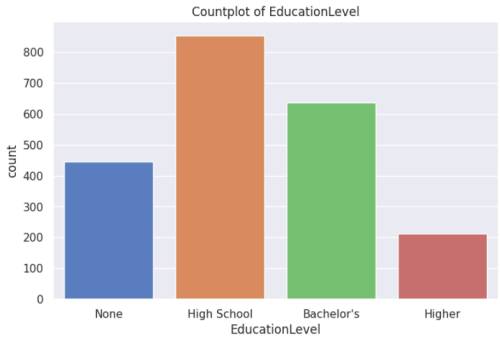
Tập dữ liệu rời rạc bao gồm các đặc trưng sau:

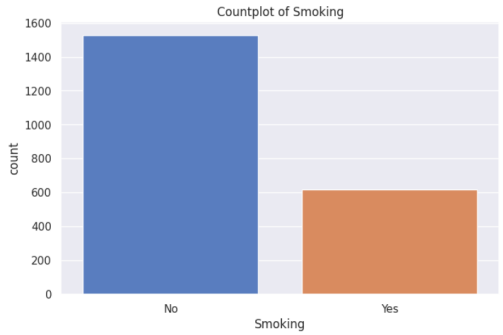
```
['BehavioralProblems', 'CardiovascularDisease', 'Confusion', 'Depression',
 'Diabetes', 'DifficultyCompletingTasks', 'Disorientation', 'EducationLevel',
 'Ethnicity', 'FamilyHistoryAlzheimers', 'Forgetfulness', 'Gender', 'HeadInjury',
 'Hypertension', 'MemoryComplaints', 'PersonalityChanges', 'Smoking']
```

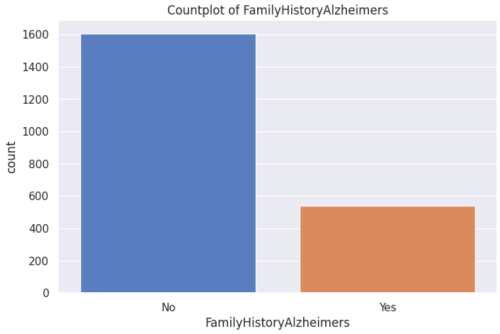
Những đặc trưng này được sử dụng để phân loại dữ liệu vào các nhóm khác nhau. Dưới đây là biểu đồ phân bổ của các giá trị trong tập dữ liệu:

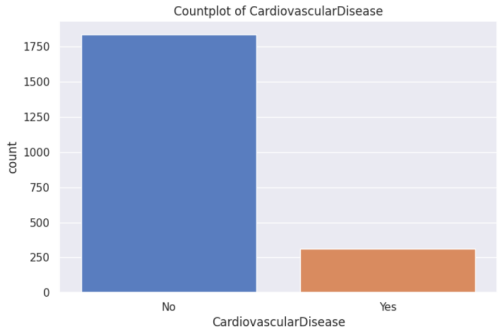


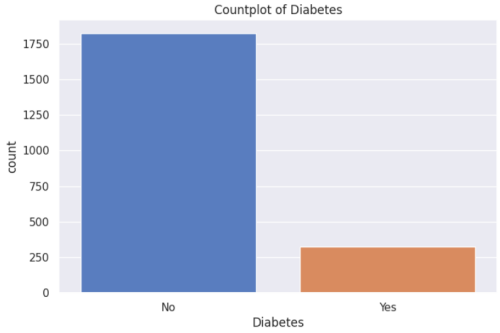


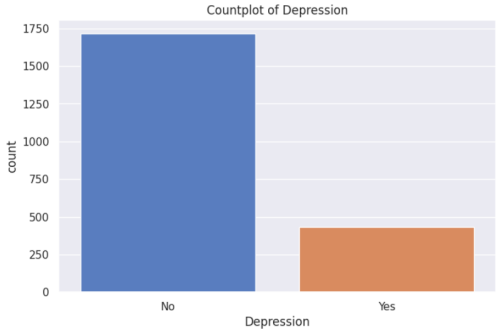


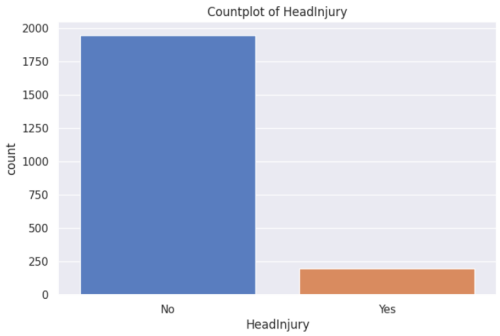


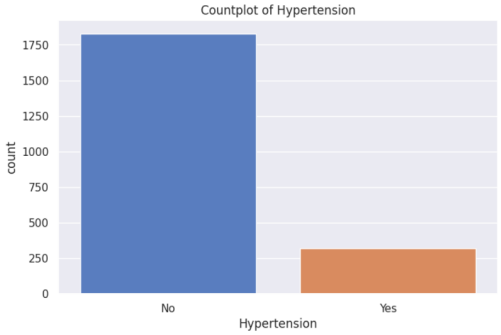

Countplot of FamilyHistoryAlzheimers
count
FamilyHistoryAlzheimers
No
Yes

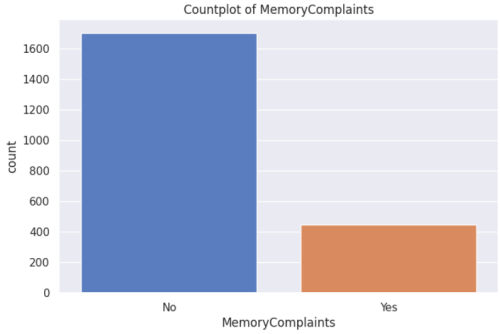

Countplot of CardiovascularDisease
count
CardiovascularDisease
No
Yes

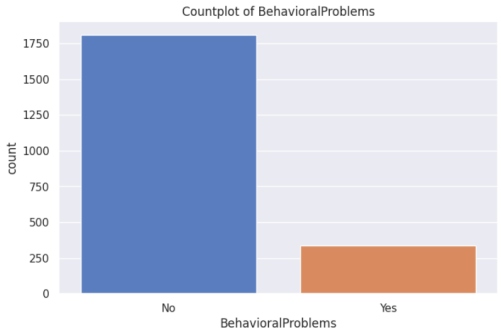

Countplot of Diabetes
count
Diabetes
No
Yes

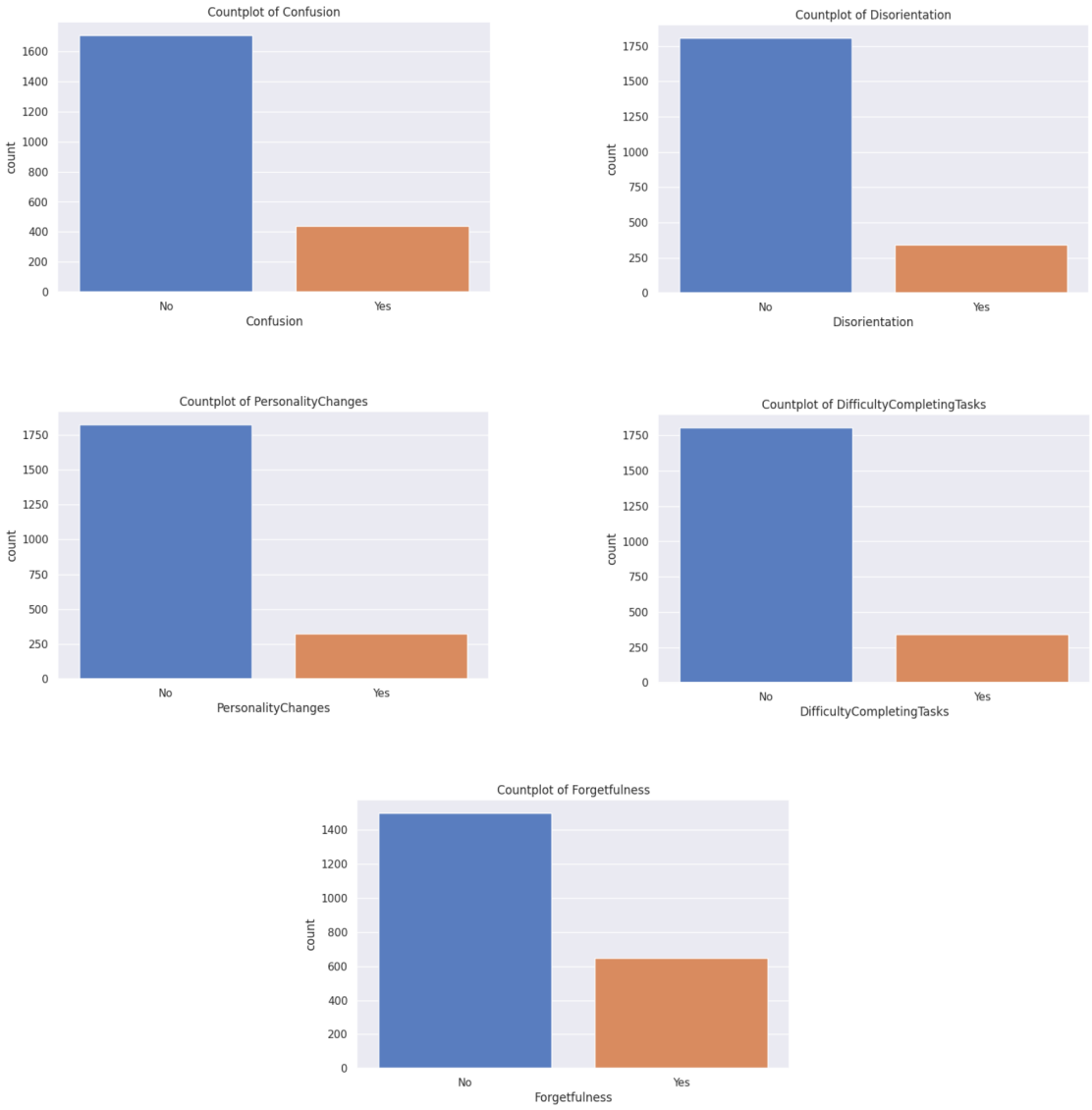

Countplot of Depression
count
Depression
No
Yes


Countplot of HeadInjury
count
HeadInjury
No
Yes


Countplot of Hypertension
count
Hypertension
No
Yes


Countplot of MemoryComplaints
count
MemoryComplaints
No
Yes


Countplot of BehavioralProblems
count
BehavioralProblems
No
Yes

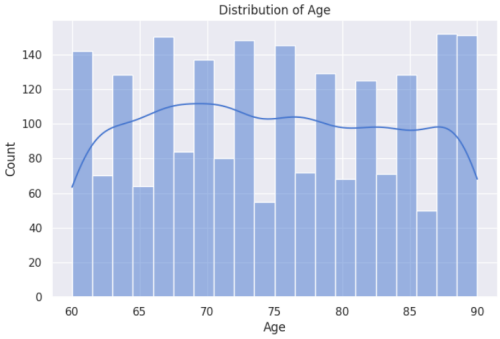


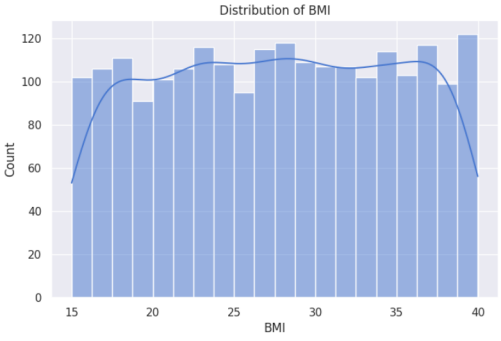
Quan sát từ việc trực quan hóa các đặc trưng rời rạc

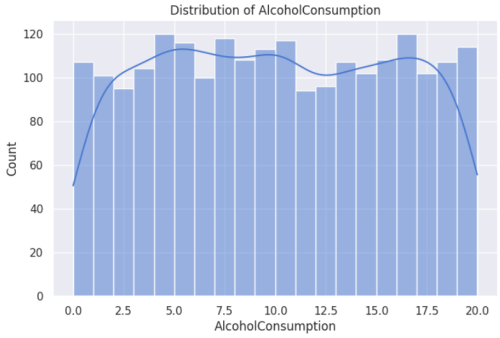
- Tổng thể, tập dữ liệu chủ yếu bao gồm những cá nhân không mắc bệnh hoặc không gặp vấn đề về sức khỏe.
- Nhóm dân số chủ yếu là người `Caucasian` (người da trắng).
- Nhóm đại diện đông nhất về mặt nhân khẩu học là những người tốt nghiệp trung học, tiếp theo là những cá nhân có bằng cử nhân.
- Đặc biệt, cả nam và nữ đều được khảo sát đồng đều trong toàn bộ tập dữ liệu.

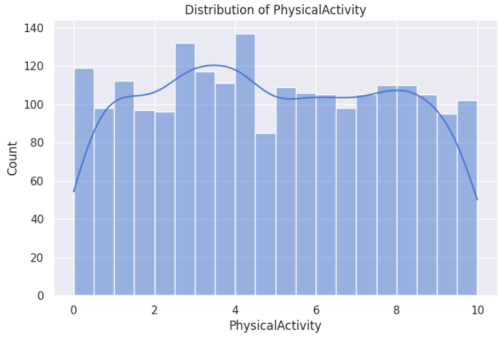
Tập dữ liệu liên tục gồm các đặc trưng còn lại:

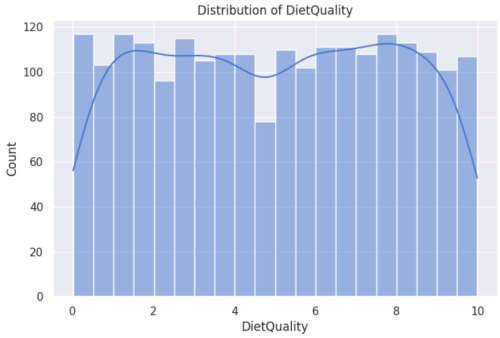
['Age', 'BMI', 'AlcoholConsumption', 'PhysicalActivity', 'DietQuality', 'SleepQuality', 'SystolicBP', 'DiastolicBP', 'CholesterolTotal', 'CholesterolLDL', 'CholesterolHDL', 'CholesterolTriglycerides', 'MMSE', 'FunctionalAssessment', 'ADL']

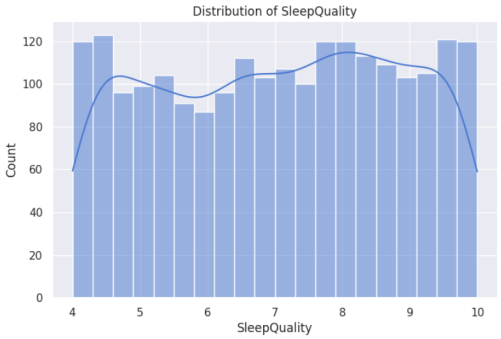


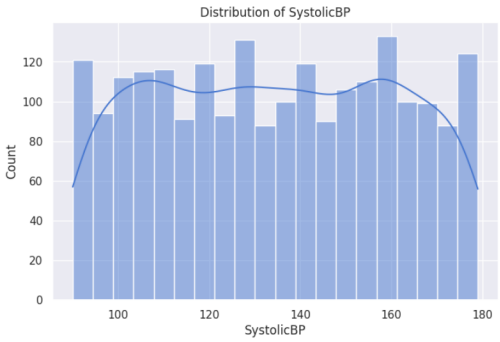


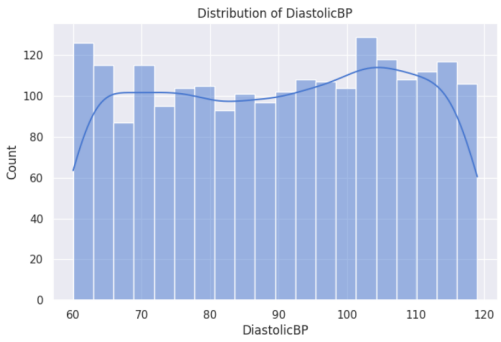


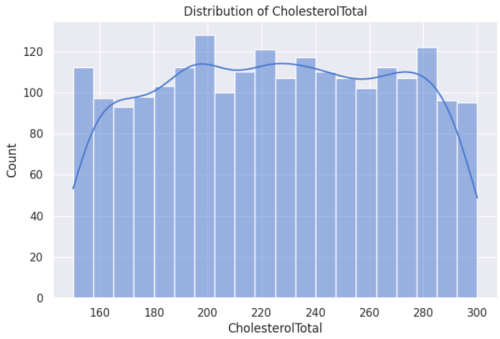


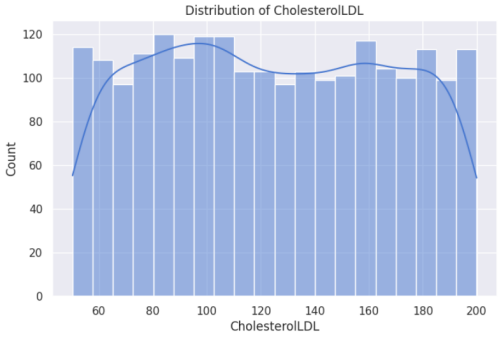


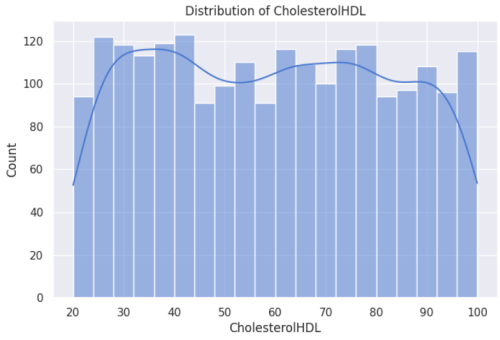


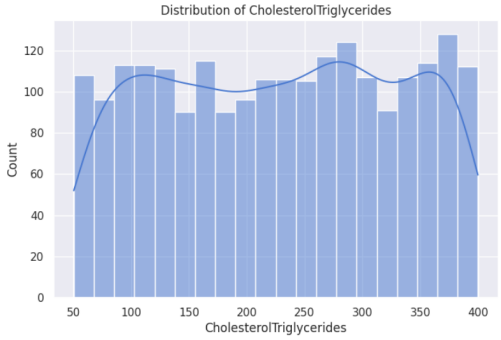

Distribution of SystolicBP
Count
SystolicBP

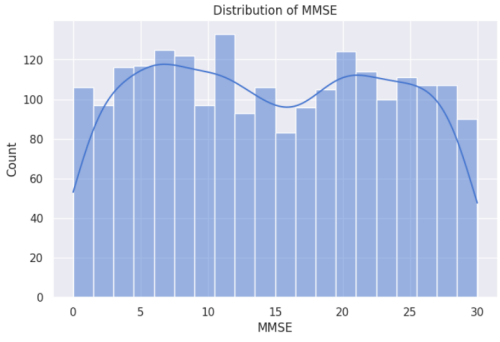

Distribution of DiastolicBP
Count
DiastolicBP

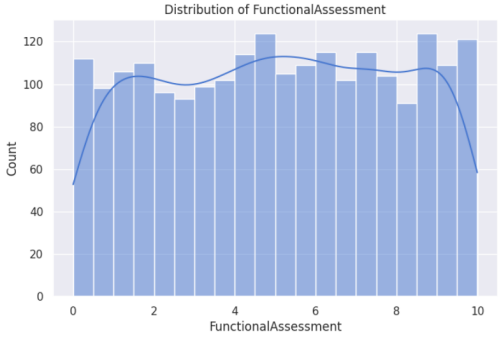

Distribution of CholesterolTotal
Count
CholesterolTotal

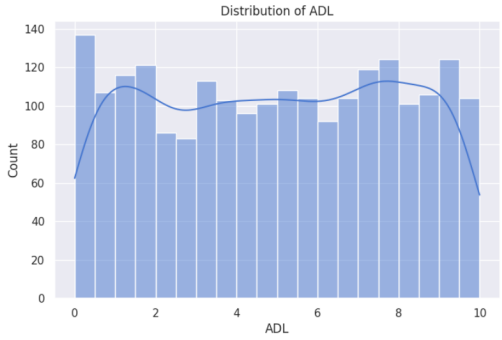

Distribution of CholesterolLDL
Count
CholesterolLDL


Distribution of CholesterolHDL
Count
CholesterolHDL


Distribution of CholesterolTriglycerides
Count
CholesterolTriglycerides


Distribution of MMSE
Count
MMSE


Distribution of FunctionalAssessment
Count
FunctionalAssessment



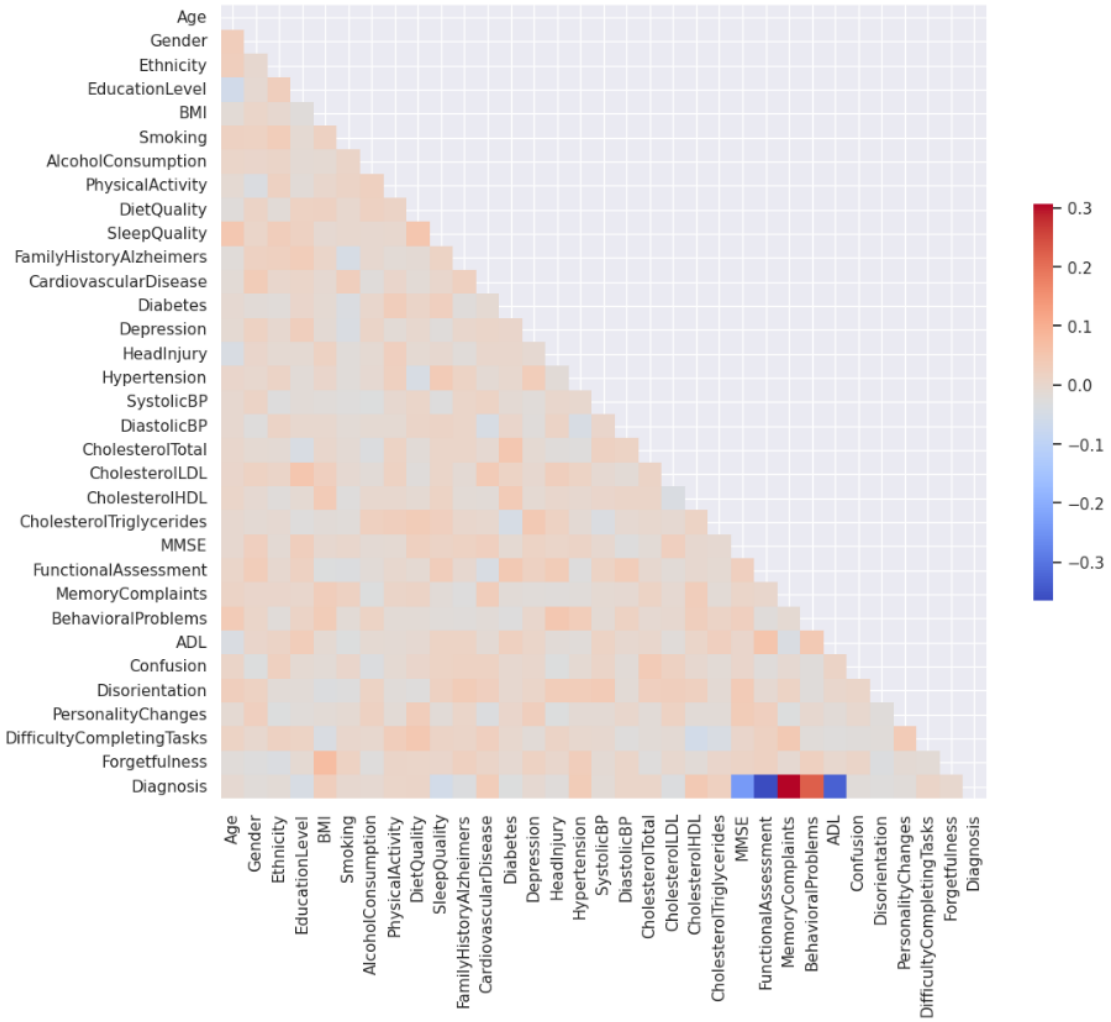
Quan sát từ biểu đồ của các đặc trưng liên tục

- Hầu hết các cột đều có phân phối khá đồng đều.
- Diểm số `MMSE` (Bài kiểm tra trạng thái tâm thần tối thiểu) có xu hướng theo phân phối hai đỉnh, cho thấy sự tồn tại của hai nhóm khác biệt trong dữ liệu.

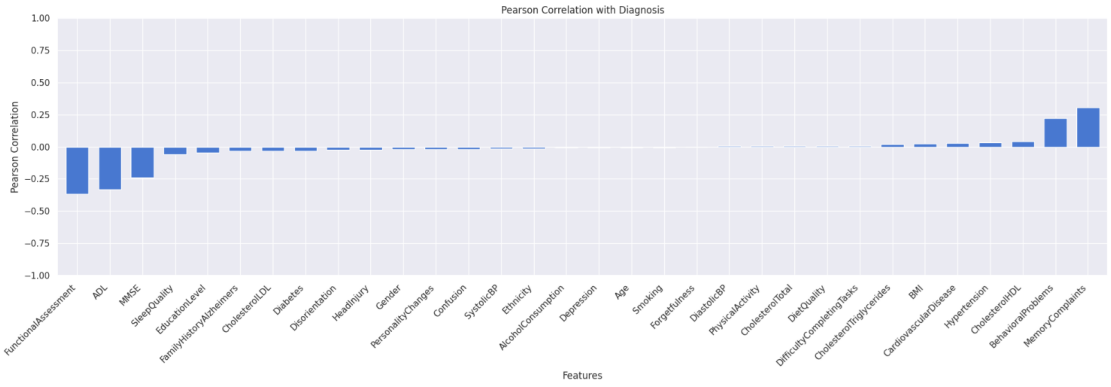
2.2 Phân tích mối quan hệ giữa các đặc trưng

Quan sát biểu đồ heatmap thể hiện mức độ tương quan giữa các đặc trưng với nhau

Biểu đồ heatmap cho thấy các đặc trưng trong dữ liệu không có mối tương quan mạnh với nhau. Tuy nhiên, có năm đặc trưng thể hiện mối tương quan với biến mục tiêu, bao gồm `MMSE`, `FunctionalAssessment`, `MemoryComplaints`, `BehaviorProblems` và `ADL`.



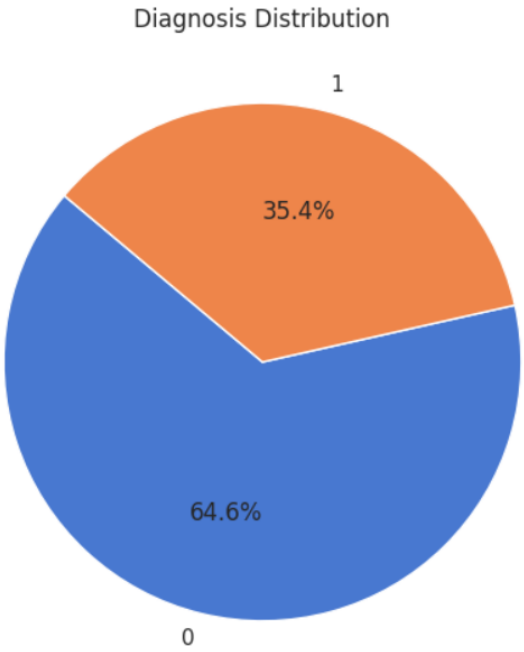
Tiếp thep, hãy cùng tính hệ số tương quan Pearson (còn gọi là Pearson's r). Đây là một thước đo mức độ mối quan hệ tuyến tính giữa hai biến. Giá trị của nó dao động từ -1 đến 1, cho biết mức độ hai biến có liên hệ tuyến tính với nhau.



Như đã quan sát, có năm đặc trưng có mối tương quan với biến mục tiêu `Diagnosis`.

- **Các đặc trưng dạng số:**
  - `FunctionalAssessment`, `ADL` (Các hoạt động sinh hoạt hằng ngày) và `MMSE` (Bài kiểm tra trạng thái tâm thần tối thiểu) đều có tương quan âm với chẩn đoán bệnh Alzheimer, với hệ số tương quan lần lượt là -0.36, -0.33 và -0.24.
  - Điều này cho thấy rằng điểm số thấp hơn trong các bài đánh giá này liên quan đến khả năng mắc bệnh Alzheimer cao hơn.
- **Các đặc trưng dạng phân loại:**
  - `BehavioralProblems` và `MemoryComplaints` có tương quan dương với chẩn đoán bệnh, với hệ số tương quan lần lượt là 0.22 và 0.30.
  - Những cá nhân có các triệu chứng này có khả năng cao hơn mắc bệnh Alzheimer, cho thấy vai trò quan trọng của những đặc trưng trong quá trình chẩn đoán.

Cuối cùng, trong tập dữ liệu quan sát, có **35.4%** người được chẩn đoán mắc bệnh Alzheimer, trong khi **64.6%** người còn lại đang ở trạng thái khỏe mạnh.



Hình 1: Phân bố chẩn đoán bệnh trong tập dữ liệu

3 Tiền xử lí dữ liệu

Trước khi huấn luyện mô hình, dữ liệu cần được xử lí để thuận lợi và đạt kết quả tốt trong quá trình huấn luyện.
Việc tiền xử lí chỉ cần thực hiện cho các biến liên tục hoặc các biến rời rạc (có hơn 2 nhóm và không có ý nghĩa thứ tự). Còn lại, biến nhị phân và các biến rời rạc (có ý nghĩa thứ tự) được giữ nguyên.

3.1 Biến liên tục

Cân bằng tỉ lệ dữ liệu có nghĩa là điều chỉnh giá trị của tất cả đặc trưng (đầu vào) sao cho tất cả giá trị các đặc trưng đó đều nằm trong một khoảng giống nhau.(thường là 0-1, hoặc là phân phối chuẩn có giá trị trung bình μ là và phương sai σ^2 là 1).
Trong Machine Learning, điều này là rất quan trọng bởi vì nhiều thuật toán Machine Learning(điển hình là SVM, KNN hoặc là Neural Network) phụ thuộc vào việc tính khoảng cách giữa các điểm để ra quyết định. Khoảng cách thường được tính bằng biểu thức Euclidean, thứ mà rất nhạy cảm với tỉ lệ của các biến/đặc trưng.
Tỉ lệ biến liên tục được hiện thực như sau:

```
columns = [
'Age','BMI','AlcoholConsumption','PhysicalActivity','DietQuality',
    'SleepQuality','SystolicBP','DiastoliP','CholesterolTotal','
    CholesterolLDL','esterolHDL','CholesterolTriglycerides','MMSE',
    'FunctionalAssessment','ADL'
]

# normalize the columns
min_max_scaler = MinMaxScaler()
df[columns] = min_max_scaler.fit_transform(df[columns])

# standardize the columns
standard_scaler = StandardScaler()
df[columns] = standard_scaler.fit_transform(df[columns])
```

Listing 1: Normalize và standardize các cột

3.2 Biến Rời rạc (không có ý nghĩa thứ tự)

Các biến/đặc trưng rời rạc có nhiều hơn 2 giá trị sẽ được nhiều thuật toán học máy (SVM, KNN, Logistic Regression,...) xem như là một biến liên tục hoặc có ý nghĩa thứ tự. Ví dụ, trong tập dữ liệu này, khi xét biến Ethnicity, các mô hình sẽ nghĩ:
African American (1) > Caucasian (0)
Asian (2) > African American (1)
Other (3) > Asian (2)
Điều này là không đúng, vì Ethnicity không có ý nghĩ so sánh như biến liên tục.Vì vậy chúng ta sẽ sử dụng One-Hot-Encoding để tiền xử lí biến Ethnicity.
Trong dữ liệu này, biến rời rạc không có ý nghĩa thứ tự duy nhất là Ethnicity và biến này được xử lí như sau:

```python
# One-Hot Encoding for 'Ethnicity'
ethnicity_encoded = pd.get_dummies(df['Ethnicity'], prefix='Ethnicity')

# Add new feature and remove feature'Ethnicity' from dataframe
df = pd.concat([df.drop(columns=['Ethnicity']), ethnicity_encoded], axis=1)
```

Listing 2: One-Hot Encoding cho biến Ethnicity

3.3 Chia dữ liệu huấn luyện và kiểm thử

Sau khi dữ liệu đã được xử lí, dữ liệu sẽ được chia ra thành 2 phần: 80% cho quá trình huấn luyện, 20% còn lại cho kiểm thử.

```python
# split data into features and target
X = df.drop(columns=['Diagnosis'])
y = df['Diagnosis']

# split data into training and testing sets
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,
    shuffle=True
)
```

Listing 3: Chia dữ liệu thành features và target, sau đó split thành train/test

4 Các mô hình dự đoán

Trong bài tập lớn này, nhóm sử dụng 4 thuật toán Machine Learning để dự đoán, 4 thuật toán đó gồm: Decision Tree, KNN, Logistic Regression, Support Vector Machine (SVM).
Đối với mỗi thuật toán, sẽ có những tham số đi kèm với thuật toán đó. Các tham số ứng với các thuật toán như sau:

Mô hình	**Tham số**	**Ý nghĩa**
Decision Tree	max_depth	Độ sâu tối đa của cây. Giới hạn số lần phân chia từ gốc tới lá; nhỏ quá → under-fit, lớn quá → over-fit.
K-Nearest Neighbors	n_neighbors	Số láng giềng được xét khi phân loại điểm mới; nhỏ quá → nhạy nhiễu, lớn quá → ranh giới mờ.
Logistic Regression	C	Độ mạnh của regularization. C lớn → phạt nhẹ → mô hình phức tạp; C nhỏ → phạt mạnh → mô hình đơn giản.
Support Vector Machine	C	Trade-off giữa margin và lỗi trên train. C lớn → margin hẹp, ít lỗi train; C nhỏ → margin rộng, cho phép lỗi train.

Bảng 4: Các tham số chính của từng mô hình

Việc điều chỉnh các tham số là rất quan trọng vì các tham số này ảnh hưởng trực tiếp đến kết quả mô hình. Để thuận tiện điều chỉnh và lựa chọn các tham số phù hợp, nhóm sử dụng GridSearchCV.
Thay vì phải thử thủ công từng giá trị kết hợp của các tham số (C, gamma, max_depth...), GridSearchCV sẽ quét toàn bộ lưới các tổ hợp do người dùng định nghĩa khi gọi hàm.
Hơn nữa, GridSearchCV cho phép đánh giá mô hình qua phương pháp K-Fold Cross Validation khi cung cấp tham số cv khi gọi hàm. Điều này giúp đánh giá mô hình khách quan và chính xác hơn. Đây là các bước hiện thực việc chọn tham số cho các mô hình:
1. Đặt tên và định nghĩa các giá trị của tham số cho mỗi mô hình

```
# define hyperparameter grids for each model
param_grids = {
    'Decision Tree': {'max_depth': [3, 5, 7, 12, None]},
    'K-Nearest Neighbors': {'n_neighbors': [3, 5, 7]},
    'Logistic Regression': {'C': [0.1, 1, 10]},
    'Support Vector Machine': {'C': [0.1, 1, 10], 'gamma': [0.1,
   1, 'scale', 'auto']},
}

# instantiate classification models with default parameters
models = {
    'Decision Tree': DecisionTreeClassifier(),
    'K-Nearest Neighbors': KNeighborsClassifier(),
    'Logistic Regression': LogisticRegression(),
    'Support Vector Machine': SVC(),
}
```

Listing 4: Định nghĩa lưới siêu tham số và khởi tạo mô hình

2. Sử dụng GridSearchCV để tìm kiếm giá trị tham số tốt nhất cho mỗi mô hình. Khi xét một mô hình, mô hình sẽ được đánh giá dựa trên 5-Fold Cross Validation (cv = 5) và tiêu chí đánh giá cho mỗi mô hình là độ chính xác (accuracy).

```
# fit models using GridSearchCV for hyperparameter tuning
for name, model in models.items():
    grid_search = GridSearchCV(
        estimator=model,
        param_grid=param_grids[name],
        cv=5,
        scoring='accuracy'
    )
    grid_search.fit(X_train, y_train)
    best_model = grid_search.best_estimator_
    y_pred = best_model.predict(X_test)
    report = classification_report(y_test, y_pred)
    print(f'{name} Classification Report:\n{report}'
          f'\nBest Parameters: {grid_search.best_params_}\n')
```

Listing 5: Điều chỉnh và lựa chọn tham số bằng GridSearch

5 Kết quả và so sánh các mô hình

Trong phần này, kết quả phân tích hiệu suất của bốn mô hình phân loại khác nhau sẽ được trình bày: Cây quyết định (Decision Tree), K-Nearest Neighbors (KNN), Hồi quy logistic (Logistic Regression), và Support vector machine (SVM). Các mô hình này được áp dụng trên tập dữ liệu để phân loại bệnh Alzheimer, và nhóm sử dụng các chỉ số đánh giá chính như *precision*, *recall*, *f1-score*, và *accuracy* để so sánh hiệu suất của từng mô hình.

5.1 Mô hình Cây quyết định (Decision Tree)

Mô hình Cây quyết định cho kết quả ấn tượng với độ chính xác cao đạt 0.93. Các chỉ số *precision* và *recall* cho lớp 0 (không mắc bệnh Alzheimer) lần lượt là 0.93 và 0.96, cho thấy mô hình có khả năng phân loại tốt nhóm không mắc bệnh. Đối với lớp 1 (mắc bệnh Alzheimer), *precision* và *recall* lần lượt là 0.92 và 0.87, chứng tỏ mô hình có thể phân biệt khá rõ các cá nhân mắc bệnh Alzheimer. F1-score cho lớp 0 đạt 0.94 và lớp 1 là 0.90, cho thấy mô hình này có sự cân bằng tốt giữa độ chính xác và khả năng phát hiện các trường hợp mắc bệnh.

Tham số tối ưu của mô hình là `max_depth = 5`, cho thấy chiều sâu của cây quyết định tối ưu là 5 để đạt được kết quả tốt nhất.

5.2 Mô hình K-Nearest Neighbors (KNN)

Kết quả từ mô hình K-Nearest Neighbors cho thấy độ chính xác thấp hơn so với Cây quyết định, chỉ đạt 0.72. Mô hình có *precision* cho lớp 0 là 0.75 và *recall* là 0.87, trong khi *precision* và *recall* cho lớp 1 lần lượt là 0.66 và 0.46, cho thấy mô hình này kém trong việc phân loại bệnh Alzheimer. F1-score của lớp 0 là 0.80, nhưng lớp 1 chỉ đạt 0.54, điều này cho thấy mô hình không phù hợp lắm với việc phát hiện các trường hợp mắc bệnh Alzheimer.

Các tham số tốt nhất cho mô hình KNN là `n_neighbors = 5`, nghĩa là số lượng láng giềng gần nhất cần thiết là 5.

5.3 Mô hình Hồi quy Logistic (Logistic Regression)

Mô hình Hồi quy logistic đạt độ chính xác là 0.83. Các chỉ số *precision* và *recall* cho lớp 0 lần lượt là 0.85 và 0.90, cho thấy mô hình có thể phân loại khá tốt nhóm không mắc bệnh. Đối với lớp 1, *precision* và *recall* là 0.79 và 0.71, điều này cho thấy mô hình có khả năng phân loại các trường hợp mắc bệnh Alzheimer ở mức độ khá. F1-score cho lớp 0 là 0.87 và lớp 1 là 0.75, cho thấy mô hình này có hiệu suất khá tốt cho cả hai lớp.

Tham số tối ưu của mô hình là `C = 1`, cho thấy đây là giá trị điều chỉnh độ mạnh của mô hình tối ưu.

5.4 Mô hình Support vector machine (SVM)

Kết quả của mô hình Máy vector hỗ trợ cho thấy độ chính xác tương đương với mô hình Hồi quy logistic, đạt 0.83. Các chỉ số *precision* và *recall* cho lớp 0 lần lượt là 0.84 và 0.92, cho thấy mô hình có khả năng phân loại tốt nhóm không mắc bệnh Alzheimer. Đối với lớp 1, *precision* và *recall* là 0.82 và 0.69, cho thấy khả năng phát hiện các trường hợp mắc bệnh Alzheimer của mô hình không được tốt như lớp 0. F1-score cho lớp 0 là 0.88 và lớp 1 là 0.75, cho thấy mô hình có hiệu suất khá cao đối với lớp không mắc bệnh nhưng có hiệu suất kém đối với lớp mắc bệnh.

Tham số tối ưu của mô hình là `C = 1` và `gamma = 'scale'`, cho thấy giá trị điều chỉnh độ mạnh của mô hình và giá trị gamma tối ưu là "scale".

5.5 Tóm tắt kết quả và so sánh

Bảng dưới đây tóm tắt kết quả của các mô hình được thử nghiệm:

Mô Hình	Accuracy	Precision (0)	Precision (1)	Recall (0)	Recall (1)
Cây quyết định (Decision Tree)	0.93	0.93	0.92	0.96	0.87
K-Nearest Neighbors (KNN)	0.72	0.75	0.66	0.87	0.46
Hồi quy Logistic (Logistic Regression)	0.83	0.85	0.79	0.90	0.71
Support vector machine (SVM)	0.83	0.84	0.82	0.92	0.69

Bảng 5: Tóm tắt kết quả của các mô hình phân loại

Kết quả cho thấy mô hình Cây quyết định (Decision Tree) có hiệu suất tốt nhất với độ chính xác cao (0.93) và sự cân bằng giữa các chỉ số precision và recall. Mô hình K-Nearest Neighbors (KNN) có hiệu suất thấp nhất, đặc biệt là đối với lớp mắc bệnh Alzheimer. Trong khi đó, cả hai mô hình Hồi quy Logistic và Máy vector hỗ trợ đều có độ chính xác tương đương (0.83), nhưng vẫn kém hơn so với Cây quyết định.

6 Code và Bộ dữ liệu

- **Link Dataset**: Alzheimer's Disease Dataset on Kaggle.
- **Link Notebook**: Kaggle Notebook for Alzheimer Prediction.

Tham khảo

[1] Eisa. (2021). *Intro to Exploratory data analysis (EDA) in Python*. https://www.kaggle.com/code/imoore/intro-to-exploratory-data-analysis-eda-in-python

[2] Yury Kashnitsky. (2021). *Exploratory Data Analysis with Pandas*. https://www.kaggle.com/code/kashnitsky/topic-1-exploratory-data-analysis-with-pandas

[3] Wei Hao Khoong. (2023). *Why Scaling Your Data Is Important*. https://medium.com/codex/why-scaling-your-data-is-important-1aff95ca97a2#:~:text=Scaling%20the%20data%20can%20help,for%20it%20to%20work%20well.