6614450042

ปัญญวัตร สุวรรณทัต

1. ตรวจสอบ Data Set Alzheimer.csv มีลักษณะข้อมูลเป็นอย่างไร type ข้อมูลแบบใด

	HN	Education	age	hypertension	heart_disease	family history	Occupation	Triglycerides	Blood Sugar	BMI	smoking_status	alz
0	11046	Secondary School	67.0	0	1	Yes	Private	Normal	228.69	36.6	formerly smoked	
1	51676	Bachelor	61.0	0	0	Yes	Freelance	High	202.21	NaN	never smoked	
2	35112	Secondary School	80.0	0	1	Yes	Private	High	105.92	32.5	never smoked	
3	67182	Bachelor	49.0	0	0	Yes	Private	Normal	171.23	34.4	smokes	
4	1665	Bachelor	79.0	1	0	Yes	Freelance	High	174.12	24.0	never smoked	
...	...	...	...	...	...	...	...	...	...	...	...	...
4095	68398	Secondary School	82.0	1	0	Yes	Freelance	High	71.97	28.3	never smoked	
4096	14180	Bachelor	13.0	0	0	No	Farmer	High	103.08	18.6	Unknown	
4097	44873	Bachelor	81.0	0	0	Yes	Freelance	Normal	125.20	40.0	never smoked	
4098	19723	Bachelor	35.0	0	0	Yes	Freelance	High	82.99	30.6	never smoked	
4099	44679	Bachelor	44.0	0	0	Yes	Officer	Normal	85.28	26.2	Unknown	

4100 rows × 12 columns

ตรวจสอบข้อมูล เช่น จำนวน row, column

```
: dataSet.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4100 entries, 0 to 4099
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   HN              4100 non-null   int64
 1   Education       4100 non-null   object
 2   age             4100 non-null   float64
 3   hypertension    4100 non-null   int64
 4   heart_disease   4100 non-null   int64
 5   family history  4100 non-null   object
 6   Occupation      4100 non-null   object
 7   Triglycerides   4100 non-null   object
 8   Blood Sugar     4100 non-null   float64
 9   BMI             3936 non-null   float64
 10  smoking_status  4100 non-null   object
 11  alzheimer       4100 non-null   int64
dtypes: float64(3), int64(4), object(5)
memory usage: 384.5+ KB
```

ตรวจสอบ type ข้อมูล

MARK: ตรวจสอบ Class Target จำนวนเหมาะสมกันหรือไม่

```
: print("Target == 1 ->", len(dataSet.loc[dataSet["alzheimer"] == 1, "alzheimer"]))
```

```
Target == 1 -> 202
```

```
: print("Target == 1 ->", len(dataSet.loc[dataSet["alzheimer"] == 0, "alzheimer"]))
```

```
Target == 1 -> 3734
```

ตรวจสอบความ Balance ของ Class target

2. ทำ Data Processing

```
dataSet = dataSet.dropna()
dataSet.isna().sum()
```

```
HN                0
Education         0
age               0
hypertension      0
heart_disease     0
family history    0
Occupation        0
Triglycerides     0
Blood Sugar       0
BMI               0
smoking_status    0
alzheimer         0
dtype: int64
```

Drop NA ของ Column BMI ทิ้ง เพราะไม่มีส่วนสูงและน้ำหนักให้คำนวณ รวมไปถึง ค่า NA ใน Column BMI นั้น มีไม่เยอะมาก ไม่น่าส่งผลกับ model เท่าที่ควร

```
dataSet['Education'].unique()
```

```
dataSet["Education"] = dataSet["Education"].replace({'Bachelor': 1, 'Secondary School': 0})
```

```
dataSet['family history'].unique()
```

```
dataSet["family history"] = dataSet["family history"].replace({'Yes': 1, 'No': 0})
```

```
dataSet['Occupation'].unique()
```

```
dataSet["Occupation"] = dataSet["Occupation"].replace({'Private': 0
                                                      , 'Freelance': 1
                                                      , "Officer":2
                                                      , "Farmer":3
                                                      , "Never_worked":4})
```

```
dataSet['Triglycerides'].unique()
```

```
dataSet['Triglycerides'] = dataSet['Triglycerides'].replace({'High': 1,'Normal': 0})
```

```
dataSet["smoking_status"].unique()
```

```
dataSet["smoking_status"] = dataSet["smoking_status"].replace({'formerly smoked': 0
                                                             , 'never smoked': 1
                                                             , "smokes":2
                                                             , "Unknown":3})
```

เปลี่ยนข้อมูล Nominal ให้อยู่ในรูป Labelencode

3. ทำการ SMOTE ข้อมูล เพื่อให้จำนวน Class Target เหมาะสมกัน
4. ทำการแบ่งชุดข้อมูลฝึกฝนและชุดข้อมูลทดสอบโดยกำหนดชุด ข้อมูลฝึกฝนมี 70% และชุดข้อมูลทดสอบมี 30%
5. ทำการฝึกฝน Model โดยเลือก 3 model ดังนี้
    a. Decision tree
    b. K nearest neighbors

c. Neural Network

โดยทำการทดของ model ละ 2 ครั้ง โดยครั้งแรกเลือก feature ทั้งหมด และครั้งที่ 2 เลือก มาเฉพาะ Feature importance 4 อันดับแรก

6. ผลการทดลอง

Model	Feature	Parameter	accuracy	precision	recall	F1
Decision tree	All	Default	0.8951	0.9	0.9	0.9
	Top 4	Default	0.8911	0.89	0.89	0.89
	Top 4	max_depth = 9	0.8380	0.84	0.84	0.84
K nearest neighbors	All	Default K = 1	0.8920	0.9	0.89	0.89
	Top 4	Default K = 1	0.8741	0.88	0.87	0.87
Neural Network	All	Default hidden_layer_sizes(64,32)	0.8995	0.9	0.9	0.9
	Top 4	Default hidden_layer_sizes(64,32)	0.8500	0.86	0.85	0.85

จากผลการทดลองจะเห็นได้ว่า Neural network แบบเลือก Feature ทั้งหมด Parameter แบบ Default ให้ความแม่นยำอยู่ที่ 89.95% ลำดับที่สองคือ Decision tree แบบเลือก Feature ทั้งหมด Parameter แบบ Default ให้ค่าความแม่นยำ อยู่ที่ 89.51% และสุดท้าย KNN เลือก Feature ทั้งหมด Parameter แบบ Default ให้ค่าความแม่นยำอยู่ที่ 89.20%