Decision Tree

ชุดข้อมูล: Psycology.xls

Source Code: 6610450951_Grading.ipynb

รูปภาพกราฟ: decision_tree.dot, decision_tree.png

(รูปภาพใช้การ render dot graphviz online เนื่องจากความบกพร่องในการ render ภาษาไทยในกราฟ)

อ่านข้อมูลจากชุดข้อมูล

Read data

```
import pandas as pd
df = pd.read_excel("Psycology.xls")
df.head()
```

[1] ✓ 0.7s Python

	เพศ	อายุ	เคยมีแฟนมาแล้ว (คน)	จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์	ระยะเวลาการนอน	นิสัยการกินอาหาร	เคยคิดฆ่าตัวตาย	จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์	ระดับความเครียดด้านการเงิน	เคยกินยานอนหลับ	ภาวะซึมเศร้า
0	Male	28	5	3	5-6 ชั่วโมง	อาหารสุขภาพ	Yes	8	3	Yes	Yes
1	Male	23	5	2	มากกว่า 8 ชั่วโมง	อาหารทั่วไป	No	10	4	No	Yes
2	Female	23	1	3	น้อยกว่า 5 ชั่วโมง	อาหารสุขภาพ	Yes	0	3	No	No
3	Female	20	5	5	มากกว่า 8 ชั่วโมง	Junkfood	Yes	2	5	No	Yes
4	Male	29	4	3	มากกว่า 8 ชั่วโมง	Junkfood	Yes	1	3	No	Yes

ตรวจสอบคอลัมน์ว่ามีคอลัมน์ใดบ้าง และแต่ละคอลัมน์ประกอบไปด้วยค่าอะไรบ้าง
ด้วยคำสั่ง .unique สำหรับทุกคอลัมน์ที่ปรากฏ

```
genders = df['เพศ'].unique()
ages = df['อายุ'].unique()
girlfriend_amount = df['เคยมีแฟนมาแล้ว  (คน)'].unique()
exercise = df['จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์'].unique()
sleep_time = df['ระยะเวลาการนอน'].unique()
food_habit = df['นิสัยการกินอาหาร'].unique()
sucide = df['เคยคิดฆ่าตัวตาย'].unique()
library_time = df['จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์'].unique()
money_status = df['ระดับความเครียดด้านการเงิน'].unique()
sleeping_pill = df['เคยกินยานอนหลับ'].unique()
is_sadness = df['ภาวะซึมเศร้า'].unique()

print('เพศ ', genders)
print('อายุ ', ages)
print('เคยมีแฟนมาแล้ว  (คน) ', girlfriend_amount)
print('จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์ ', exercise)
print('ระยะเวลาการนอน', sleep_time)
print('นิสัยการกินอาหาร ', food_habit)
print('เคยคิดฆ่าตัวตาย ', sucide)
print('จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์ ', library_time)
print('ระดับความเครียดด้านการเงิน ', money_status)
print('เคยกินยานอนหลับ ', sleeping_pill)
print('ภาวะซึมเศร้า', is_sadness)
```

[3] ✓ 0.0s

```
เพศ  ['Male' 'Female']
อายุ  [28 23 20 29 31 24 33 25 19 34 21 30 32 26 22 27 18]
เคยมีแฟนมาแล้ว  (คน)  [5 1 4 2 3 '?']
จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์  [3 2 5 4 '?' 1]
ระยะเวลาการนอน ['5-6 ชั่วโมง' 'มากกว่า 8 ชั่วโมง' 'น้อยกว่า 5 ชั่วโมง' '7-8 ชั่วโมง']
นิสัยการกินอาหาร  ['อาหารสุขภาพ' 'อาหารทั่วไป' 'Junkfood']
เคยคิดฆ่าตัวตาย  ['Yes' 'No']
จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์  [8 10 0 2 1 3 11 12 9 7 4 6 '?' 5]
ระดับความเครียดด้านการเงิน  [3 4 5 '?' 1 2]
เคยกินยานอนหลับ  ['Yes' 'No']
ภาวะซึมเศร้า ['Yes' 'No']
```

ข้อมูลที่ได้จากกลุ่มข้อมูลตัวอย่าง

- เพศ
    - ชาย (Male)
    - หญิง (Female)
- อายุ
    - ตัวเลขจำนวนเต็มที่อยู่ในช่วง 18 – 34 ปี
- เคยมีแฟนมาแล้ว (คน)
    - ตัวเลขจำนวนเต็มที่อยู่ในช่วง 1 – 5
    - พบค่าที่หายไป “?”
- จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์
    - ตัวเลขจำนวนเตาที่อยู่ในช่วง 1 – 5
    - พบค่าที่หายไป “?”
- ระยะเวลาการนอน
    - น้อยกว่า 5 ชั่วโมง
    - 5-6 ชั่วโมง
    - 7-8 ชั่วโมง
    - มากกว่า 8 ชั่วโมง
- นิสัยการกินอาหาร
    - อาหารสุขภาพ
    - อาหารทั่วไป
    - Junkfood
- เคยคิดฆ่าตัวตาย
    - Yes
    - No
- จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์
    - จำนวนเต็มที่อยู่ในช่วง 0 – 12
    - พบค่าที่หายไป “?”
- ระดับความเครียดด้านการเงิน
    - จำนวนเต็มที่อยู่ในช่วง 1 – 5
    - พบค่าที่หายไป “?”
- เคยกินยานอนหลับ
    - Yes
    - No

และผลลัพท์ของการเป็นโรคซึมเศร้า

- Yes
- No

ทดลองตัดชุดข้อมูลตัวอย่างทุกแถวที่มีสิ่งที่ไม่ทราบค่าปรากฏ

```
    #Before
    print(f"Data size before cut is {df.shape[0]}")

    #cut
    df = df[~df.isin(['?']).any(axis=1)]

    #After
    print(f"Data size after cut is {df.shape[0]}")

[4] ✓ 0.0s
...  Data size before cut is 502
     Data size after cut is 467
```

พบว่ามีชุดข้อมูลที่ถูกตัดออกไปทั้งหมด 35 ตัวอย่างจาก 502 ตัวอย่าง

Map ชุดข้อมูลทั้งตารางให้อยู่ในรูปตัวเลข เพื่อสะดวกต่อการนำไปคิดต่อ

```
df['เพศ'] = df['เพศ'].replace({'Male': 1, 'Female': 0})

#'อายุ' is no need for mapping
#'เคยมีแฟนมาแล้ว  (คน)' is no need for mapping
#'จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์' is no need for mapping

df['ระยะเวลาการนอน'] = df['ระยะเวลาการนอน'].replace({
    '5-6 ชั่วโมง': 5.5,
    'มากกว่า 8 ชั่วโมง': 8.5,
    'น้อยกว่า 5 ชั่วโมง': 4.5,
    '7-8 ชั่วโมง': 7.5
})

df['นิสัยการกินอาหาร'] = df['นิสัยการกินอาหาร'].replace({
    'อาหารสุขภาพ': 0,
    'อาหารทั่วไป': 1,
    'Junkfood': 2
})

df['เคยคิดฆ่าตัวตาย'] = df['เคยคิดฆ่าตัวตาย'].replace({'Yes': 1, 'No': 0})

#'จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์' is no need for mapping
#'ระดับความเครียดด้านการเงิน' is no need for mapping

df['เคยกินยานอนหลับ'] = df['เคยกินยานอนหลับ'].replace({'Yes': 1, 'No': 0})
df['ภาวะซึมเศร้า'] = df['ภาวะซึมเศร้า'].replace({'Yes': 1, 'No': 0})
```

กำหนดให้การ Map ข้อมูลเป็นดังนี้

- เพศ
    - ชาย -> 1
    - หญิง -> 0
- ระยะเวลาการนอน จะใช้การประมาณค่า
    - น้อยกว่า 5 ชั่วโมง -> 4.5
    - 5-6 ชั่วโมง -> 5.5
    - 7-8 ชั่วโมง -> 7.5
    - มากกว่า 8 ชั่วโมง -> 8.5
- นิสัยการกินอาหาร
    - อาหารสุขภาพ -> 0
    - อาหารทั่วไป -> 1
    - Junkfood -> 2

- เคยคิดฆ่าตัวตาย
  - Yes -> 1
  - No -> 0
- เคยกินยานอนหลับ
  - Yes -> 1
  - No -> 0

<u>และ Map ข้อมูลผลลัพท์เป็น</u>

โรคซึมเศร้า

- Yes -> 1
- No -> 0

แบ่งเป็น Features และ Target จากนั้นทำ Train Test Split

โดยใช้ DecisionTreeClassifier จาก Library sklearn.tree

```
X = df.iloc[:, :-1]  # Features
y = df.iloc[:, -1] # Target
```
✓ 0.0s

Do 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)
```
✓ 1.6s

DecisionTreeClassifier

```
from sklearn.tree import DecisionTreeClassifier

DTreeClassifier = DecisionTreeClassifier(random_state=42)
DTreeClassifier.fit(X_train, y_train)
```
✓ 0.1s

DecisionTreeClassifier

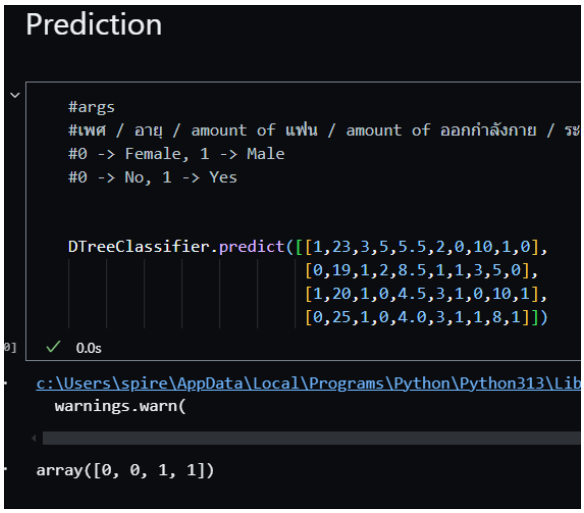
DecisionTreeClassifier(random_state=42)

```
DTreeClassifier.score(X_test, y_test)
```
✓ 0.0s

0.7978723404255319

และใช้ .score เพื่อดูค่าความแม่นยำในการ Predict Test set

ทดสอบการ Predict ค่า



ผู้จัดทำได้เตรียมชุดทดสอบไว้ 4 กรณี

กรณีทดสอบที่ 1

- เพศ -> ชาย (Male) -> 1
- อายุ -> 23 ปี
- เคยมีแฟนมาแล้ว -> 3 คน
- จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์ -> 5 ครั้ง
- ระยะเวลาการนอน -> 5.5 ชั่วโมง
- นิสัยการกินอาหาร -> อาหารทั่วไป
- เคยคิดฆ่าตัวตาย -> No
- จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์ -> 10 ครั้ง
- ระดับความเครียดด้านการเงิน -> ระดับ 1
- เคยกินยานอนหลับ -> No

ผลลัพท์ : ไม่เป็นโรคซึมเศร้า

กรณีทดสอบที่ 2

- เพศ -> หญิง (Female) -> 0
- อายุ -> 19 ปี
- เคยมีแฟนมาแล้ว -> 1 คน
- จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์ -> 2 ครั้ง
- ระยะเวลาการนอน -> 8.5 ชั่วโมง
- นิสัยการกินอาหาร -> อาหารสุขภาพ
- เคยคิดฆ่าตัวตาย -> Yes
- จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์ -> 3 ครั้ง
- ระดับความเครียดด้านการเงิน -> ระดับ 5
- เคยกินยานอนหลับ -> No

ผลลัพท์ : ไม่เป็นโรคซึมเศร้า

กรณีทดสอบที่ 3

- เพศ -> ชาย (Male) -> 1
- อายุ -> 20 ปี
- เคยมีแฟนมาแล้ว -> 1 คน
- จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์ -> 0 ครั้ง
- ระยะเวลาการนอน -> 4.5 ชั่วโมง
- นิสัยการกินอาหาร -> Junkfood
- เคยคิดฆ่าตัวตาย -> Yes
- จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์ -> 0 ครั้ง
- ระดับความเครียดด้านการเงิน -> ระดับ 10
- เคยกินยานอนหลับ -> Yes

ผลลัพท์ : เป็นโรคซึมเศร้า

กรณีทดสอบที่ 4

- เพศ -> หญิง (Female) -> 0
- อายุ -> 25 ปี
- เคยมีแฟนมาแล้ว -> 1 คน
- จำนวนครั้งที่ไปออกกำลังกายต่อสัปดาห์ -> 0 ครั้ง
- ระยะเวลาการนอน -> 4.0 ชั่วโมง
- นิสัยการกินอาหาร -> Junkfood
- เคยคิดฆ่าตัวตาย -> Yes
- จำนวนชั่วโมงที่เข้าห้องสมุดในหนึ่งสัปดาห์ -> 1 ครั้ง
- ระดับความเครียดด้านการเงิน -> ระดับ 8
- เคยกินยานอนหลับ -> Yes

ผลลัพท์ : เป็นโรคซึมเศร้า

ผลลัพธ์และข้อสรุป

- ประสิทธิภาพของโมเดลเมื่อทดสอบบน test set อยู่ที่ 0.79 หรือ 79%
- ความสำคัญของ feature ต่างๆ

```
    DTreeClassifier.feature_importances_
✓ 0.0s
array([0.00357588, 0.14238101, 0.25476632, 0.14105337, 0.0386228 ,
       0.03256605, 0.19099132, 0.05791648, 0.12286167, 0.0152651 ])
```

โดยใช้คำสั่ง .feature_importances_ เพื่อแสดงค่าฟีเจอร์ที่สำคัญที่สุดนั่นคือจำนวนแฟนที่เคยมีมาแล้ว ซึ่งมีค่าที่สูงที่สุด คือ 0.25476632 ซึ่งส่งผลสำคัญที่สุดต่อการจัดกลุ่มของข้อมูล

รูปภาพกราฟต้นไม้ (สามารถดูรูปภาพฉบับเต็มได้จากไฟล์แนบ)