รายละเอียดผลการทดลอง

6610450871 นายชนพัฒน์ โชติกุลรัตน์ หมู่ 200

Model = Decision Tree
Dataset = IndiaWeather

วิธีที่ 1

1. เริ่มต้นด้วยการอ่านข้อมูลจากไฟล์ Excel

2. ค่าที่ผิดปกติหรือสูญหายภายในชุดข้อมูลได้รับการจัดการโดยการแทนที่ด้วยค่า np.nan โดยใช้ไลบรารี NumPy วิธีนี้ช่วยให้สามารถจัดการกับข้อมูลที่สูญหายได้อย่างเหมาะสมและป้องกันข้อผิดพลาดระหว่างการฝึกฝน

3. features ที่ใช้คือ features ทั้งหมดที่มีอยู่ในชุดข้อมูลถูกนำมาใช้สำหรับการฝึกฝนการทำนาย

อุณหภูมิ	ความชื้น	ปริมาณ PM2.5	ปริมาณ PM10	ปริมาณไนโตรเจน	ปริมาณซัลเฟอร์	ปริมาณคาร์บอน	ระยะห่างจากโรงงาน	ความหนาแน่นประชากร

4. จากนั้นชุดข้อมูลจะถูกแบ่งออกเป็นชุดฝึกฝนและชุดทดสอบโดยใช้ฟังก์ชัน train_test_split การแบ่งนี้ช่วยให้มั่นใจได้ว่าประสิทธิภาพของแบบจำลองสามารถประเมินได้จากข้อมูลที่มองไม่เห็น มีการใช้ test_size เท่ากับ 0.2 ซึ่งหมายความว่า 20% ของข้อมูลถูกสงวนไว้สำหรับการทดสอบ ในขณะที่ 80% ที่เหลือใช้สำหรับการฝึกฝน มีการตั้งค่า random_state เป็น 1 เพื่อให้แน่ใจว่าผลลัพธ์สามารถทำซ้ำได้ การตั้งค่าสถานะแบบสุ่มทำให้มั่นใจได้ว่าการแบ่งข้อมูลจะสอดคล้องกันหากรันโค้ดหลายครั้ง ทำให้สามารถเปรียบเทียบระหว่างการรันหรือการกำหนดค่าแบบจำลองต่างๆ ได้อย่างยุติธรรม

5. แบบจำลอง DecisionTreeClassifier ถูกนำมาใช้สำหรับการทำนาย แบบจำลองนี้ใช้โครงสร้างแบบต้นไม้เพื่อทำการตัดสินใจตามคุณลักษณะอินพุต อีกครั้ง มีการใช้ random_state เท่ากับ 1 เพื่อให้ได้ผลลัพธ์ที่สอดคล้องกันในการรันหลายครั้ง

6. การทำนายและการประเมินผล (ส่วนนี้ต้องการผลลัพธ์จริงเพื่อให้สมบูรณ์) จากนั้นใช้ DecisionTreeClassifier ที่ผ่านการฝึกฝนแล้วเพื่อทำนายตัวแปรเป้าหมายบนชุดทดสอบที่แยกไว้ ประสิทธิภาพของแบบจำลองได้รับการประเมินโดยใช้เมตริกที่เหมาะสม (เช่น ความแม่นยำ, ความเที่ยงตรง, ความระลึก, คะแนน F1) ผลลัพธ์ของการทำนายและการประเมินผลมีดังนี้:

```
Accuracy: 0.23
                precision    recall  f1-score   support

            ดี       0.34      0.28      0.31        47
       ปานกลาง       0.28      0.23      0.25        30
อันตรายต่อสุขภาพ       0.00      0.00      0.00        11
           แย่       0.12      0.25      0.16        12

      accuracy                           0.23       100
     macro avg       0.18      0.19      0.18       100
  weighted avg       0.26      0.23      0.24       100
```

วิธีที่ 2

1. มีการจัดการค่าที่หายไปโดยใช้วิธีการแทนที่ค่าที่คำนึงถึงป้ายกำกับ (label-aware imputation) แทนที่จะแทนที่ค่าที่หายไปด้วยค่าเฉลี่ยของทั้งคอลัมน์ เราใช้วิธีการที่มีความละเอียดอ่อนมากขึ้น สำหรับแต่ละคุณลักษณะที่เป็นตัวเลข ค่าที่หายไปจะถูกแทนที่ด้วยค่าเฉลี่ยที่คำนวณ เฉพาะเจาะจงสำหรับแต่ละป้ายกำกับคุณภาพอากาศ วิธีนี้คำนึงถึงความแตกต่างที่อาจเกิดขึ้นในการกระจายของคุณลักษณะในแต่ละระดับคุณภาพอากาศ ส่งผลให้การแทนที่ค่ามีความแม่นยำและสอดคล้องกับบริบทมากขึ้น วิธีนี้ดีกว่าการใช้ค่าเฉลี่ยของทั้งคอลัมน์ เพราะรักษาความสัมพันธ์ระหว่างคุณลักษณะและตัวแปรเป้าหมาย (คุณภาพอากาศ) การแทนที่ค่าที่คำนึงถึงป้ายกำกับนี้ช่วยรักษาความสมบูรณ์ของข้อมูลและป้องกันการเกิดอคติเนื่องจากข้อมูลที่หายไป

ค่า Accuracy เพิ่มขึ้น 8 %

```
Accuracy: 0.31
                  precision    recall  f1-score   support

              ดี       0.42      0.38      0.40        47
         ปานกลาง       0.37      0.37      0.37        30
อันตรายต่อสุขภาพ       0.00      0.00      0.00        11
             แย่       0.11      0.17      0.13        12

        accuracy                           0.31       100
       macro avg       0.22      0.23      0.22       100
    weighted avg       0.32      0.31      0.31       100
```

วิธีที่ 3

เพื่อเตรียมข้อมูลให้พร้อมสำหรับการฝึกฝนแบบจำลอง มีการใช้เทคนิคการแปลงและปรับขนาดข้อมูลสองเทคนิค ตัวแปรเป้าหมาย "คุณภาพอากาศ" ซึ่งเป็นข้อมูลประเภทหมวดหมู่ ถูกแปลงเป็นตัวเลขโดยใช้ LabelEncoder ขั้นตอนนี้จำเป็นสำหรับอัลกอริทึมการเรียนรู้ของเครื่องหลายๆ ตัวที่ต้องการอินพุตเป็นตัวเลข นอกจากนี้ คุณลักษณะที่เป็นตัวเลขทั้งหมดถูกปรับขนาดโดยใช้ StandardScaler เพื่อให้มีค่าเฉลี่ยเป็น 0 และส่วนเบี่ยงเบนมาตรฐานเป็น 1 การปรับขนาดนี้ช่วยป้องกันไม่ให้คุณลักษณะที่มีช่วงค่าขนาดใหญ่ครอบงำคุณลักษณะที่มีช่วงค่าขนาดเล็ก และยังช่วยเพิ่มประสิทธิภาพของแบบจำลองอีกด้วย

ค่า Accuracy ไม่เพิ่มขึ้น

วิธีที่ 4

เนื่องจากค่า ppb (ส่วนในพันล้านส่วน) และ ppm (ส่วนในล้านส่วน) มีหน่วยต่างจาก μg/m³ (ไมโครกรัมต่อลูกบาศก์เมตร) จึงจำเป็นต้องแปลงหน่วยของข้อมูลความเข้มข้นของก๊าซไนโตรเจนไดออกไซด์ (NO2), ซัลเฟอร์ไดออกไซด์ (SO2) และคาร์บอนมอนอกไซด์ (CO) ให้เป็น μg/m³ เพื่อให้สอดคล้องกัน การแปลงนี้ใช้ค่าคงที่น้ำหนักโมเลกุลของก๊าซแต่ละชนิด และใช้สมมติฐานเกี่ยวกับอุณหภูมิและความดันมาตรฐาน (STP) ในการคำนวณปริมาตรโมลาร์ของก๊าซ โดยใช้ฟังก์ชัน ppb_to_ugm3 และ ppm_to_ugm3 ที่ได้กำหนดขึ้น การแปลงหน่วยนี้ช่วยให้มั่นใจได้ว่าข้อมูลที่ใช้ในการวิเคราะห์และสร้างแบบจำลองมีความถูกต้องและสอดคล้องกันมากขึ้น

ค่า Accuracy ไม่เพิ่มขึ้น

วิธีที่ 5

กรองข้อมูล PM2.5 ที่ไม่สอดคล้อง เพื่อเพิ่มความถูกต้องของแบบจำลอง
ได้ทำการกรองข้อมูลโดยใช้ค่า PM2.5 โดยกำหนดช่วงค่า PM2.5 และป้ายกำกับคุณภาพอากาศที่สอดคล้องกันดังนี้: 0-37 (ดี), 38-50 (ปานกลาง), 51-90 (แย่) และ > 91 (อันตราย) จากนั้นจึงสร้าง dataframe ใหม่ชื่อ "pm25_filtered" เพื่อจัดกลุ่มข้อมูล PM2.5 และกรองข้อมูลโดยเลือกเฉพาะแถวที่ค่า "pm25_filtered" ตรงกับป้ายกำกับ "คุณภาพอากาศ" มีข้อมูลจำนวน 323 แถว ที่ถูกกรองออก เนื่องจากค่า PM2.5 ไม่สอดคล้องกับป้ายกำกับคุณภาพอากาศ

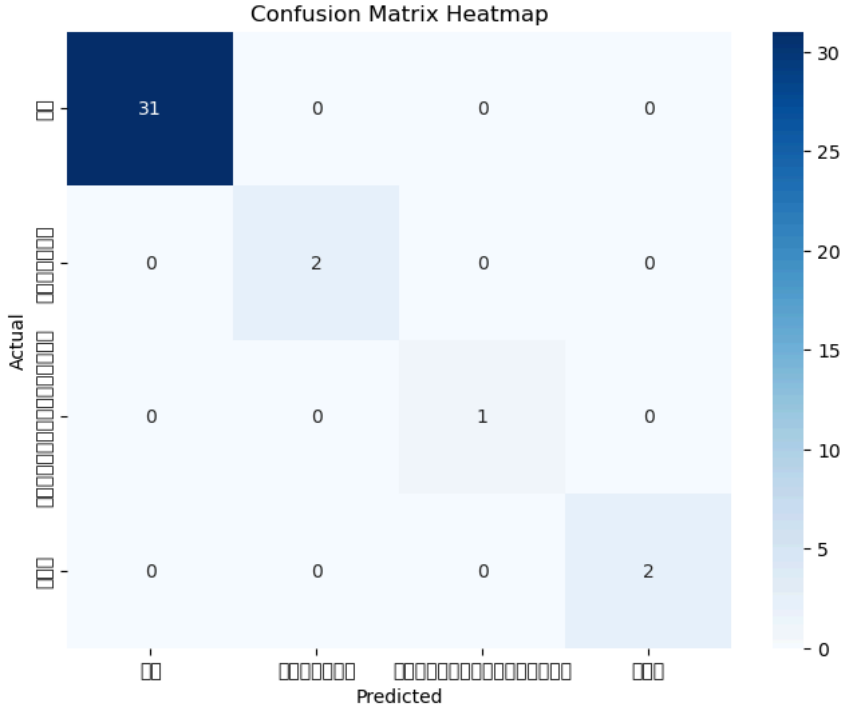
ที่มา : กรมควบคุมโรค

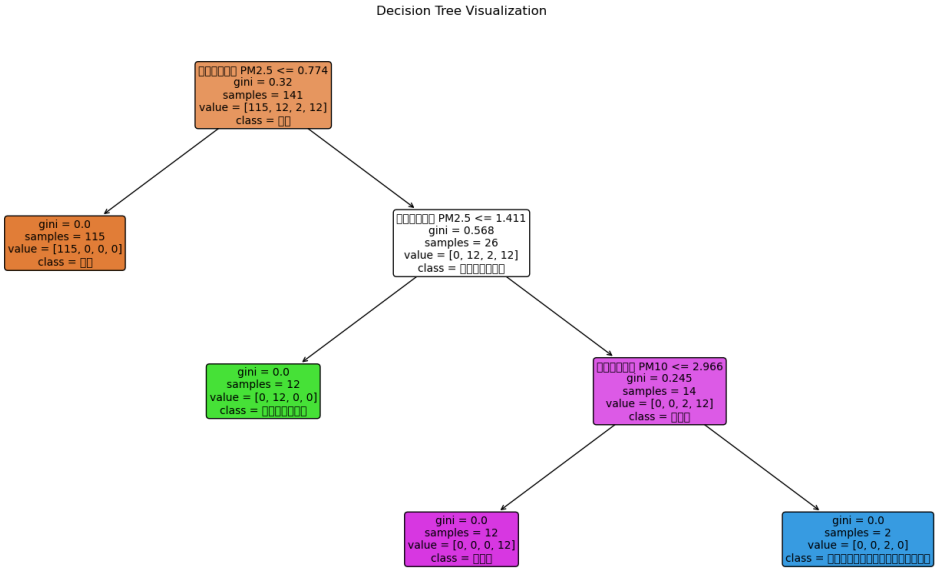
ค่า Accuracy = 1.0

```
Accuracy: 1.0
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        31
           1       1.00      1.00      1.00         2
           2       1.00      1.00      1.00         1
           3       1.00      1.00      1.00         2

    accuracy                           1.00        36
   macro avg       1.00      1.00      1.00        36
weighted avg       1.00      1.00      1.00        36
```


Confusion Matrix Heatmap
Actual
Predicted
31
0
0
0
0
2
0
0
0
0
1
0
0
0
0
2
30
25
20
15
10
5
0


Decision Tree Visualization
PM2.5 <= 0.774
gini = 0.32
samples = 141
value = [115, 12, 2, 12]
gini = 0.0
samples = 115
value = [115, 0, 0, 0]
PM2.5 <= 1.411
gini = 0.568
samples = 26
value = [0, 12, 2, 12]
gini = 0.0
samples = 12
value = [0, 12, 0, 0]
PM10 <= 2.966
gini = 0.245
samples = 14
value = [0, 0, 2, 12]
gini = 0.0
samples = 12
value = [0, 0, 0, 12]
gini = 0.0
samples = 2
value = [0, 0, 2, 0]