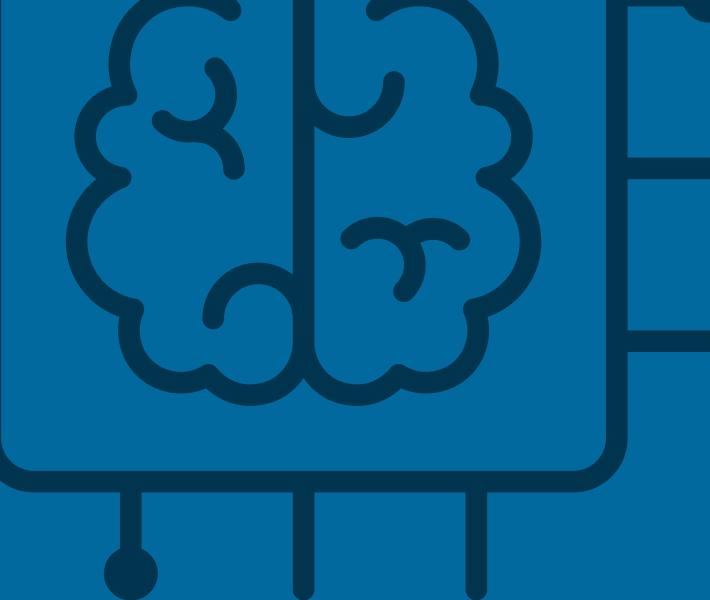
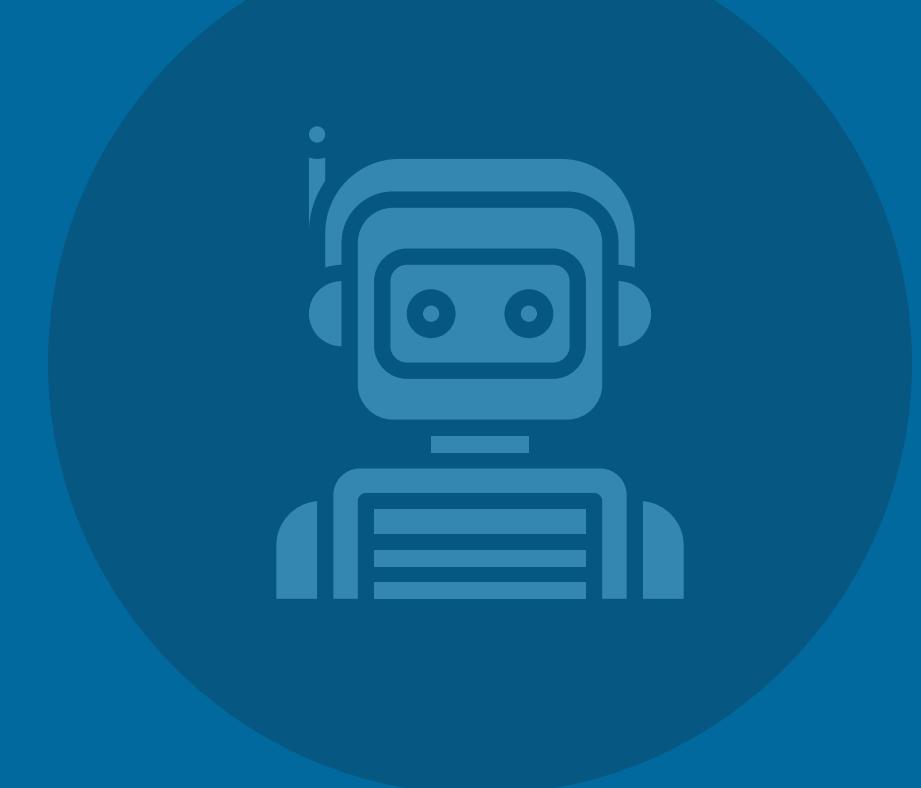
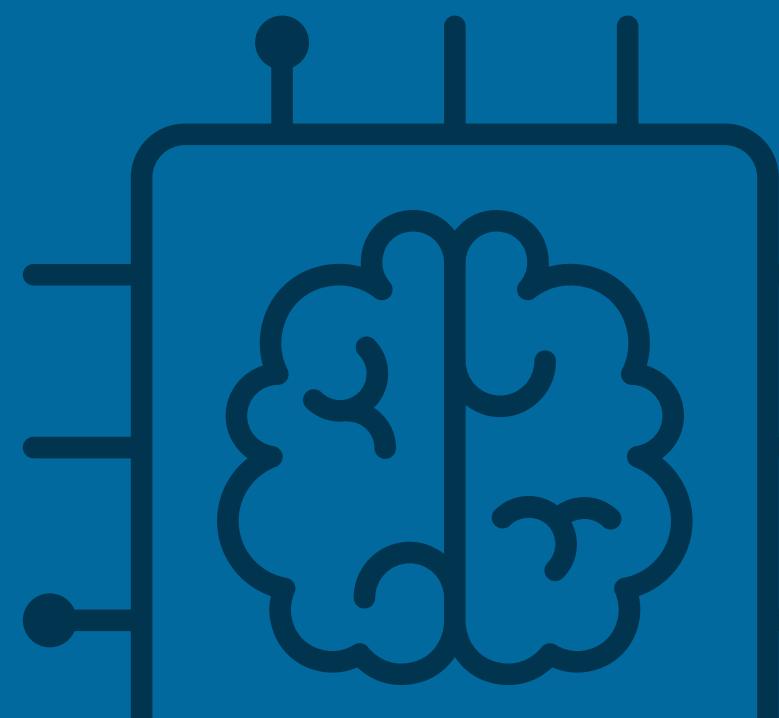
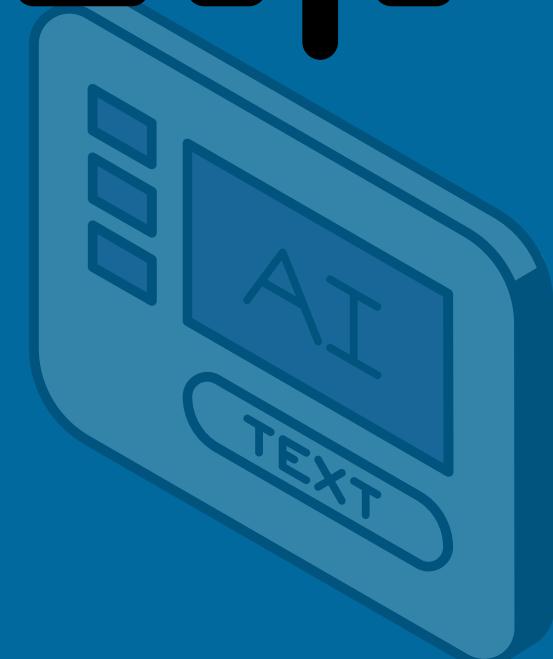
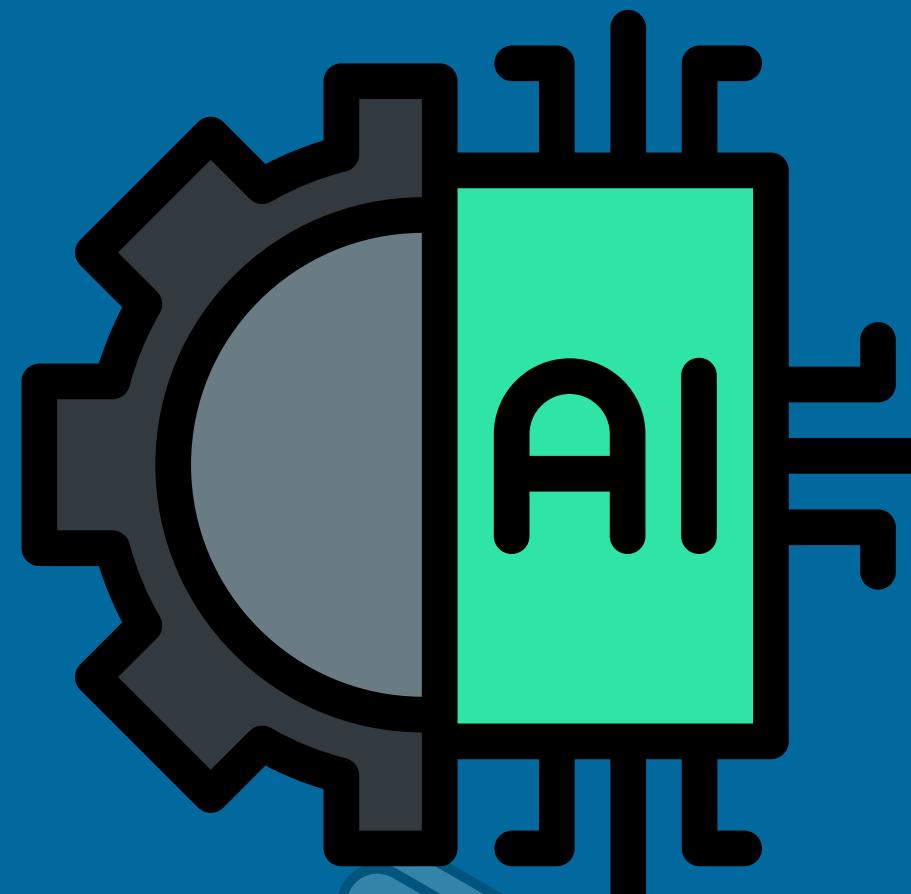
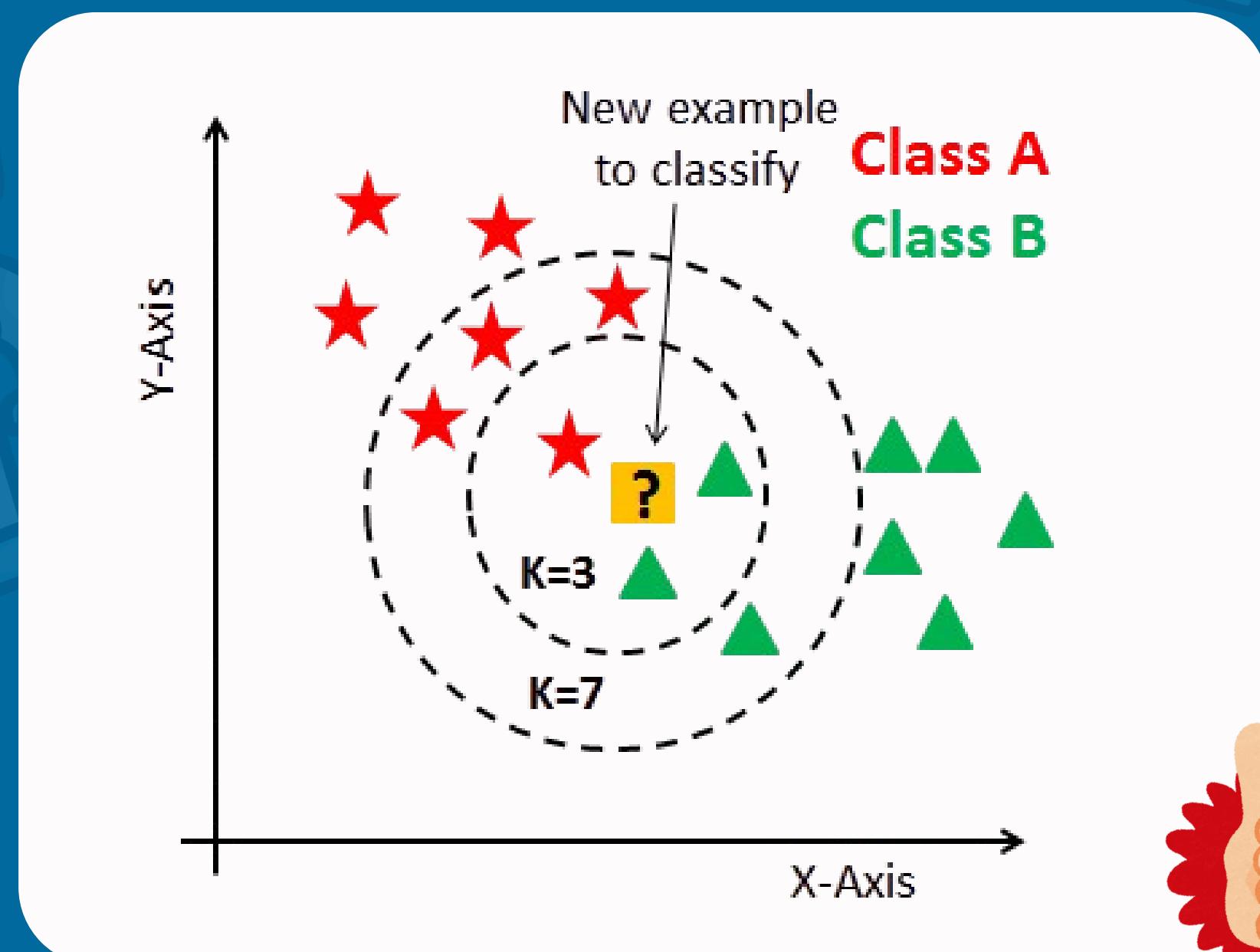


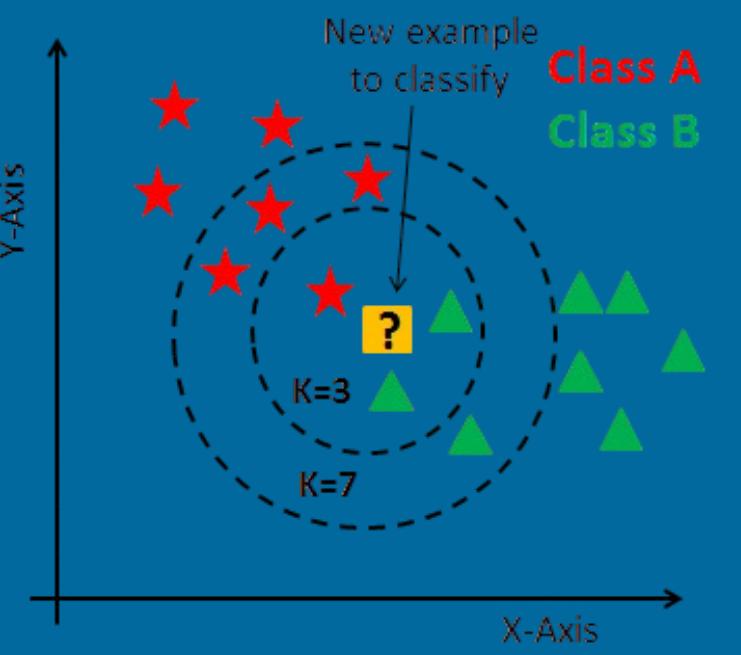
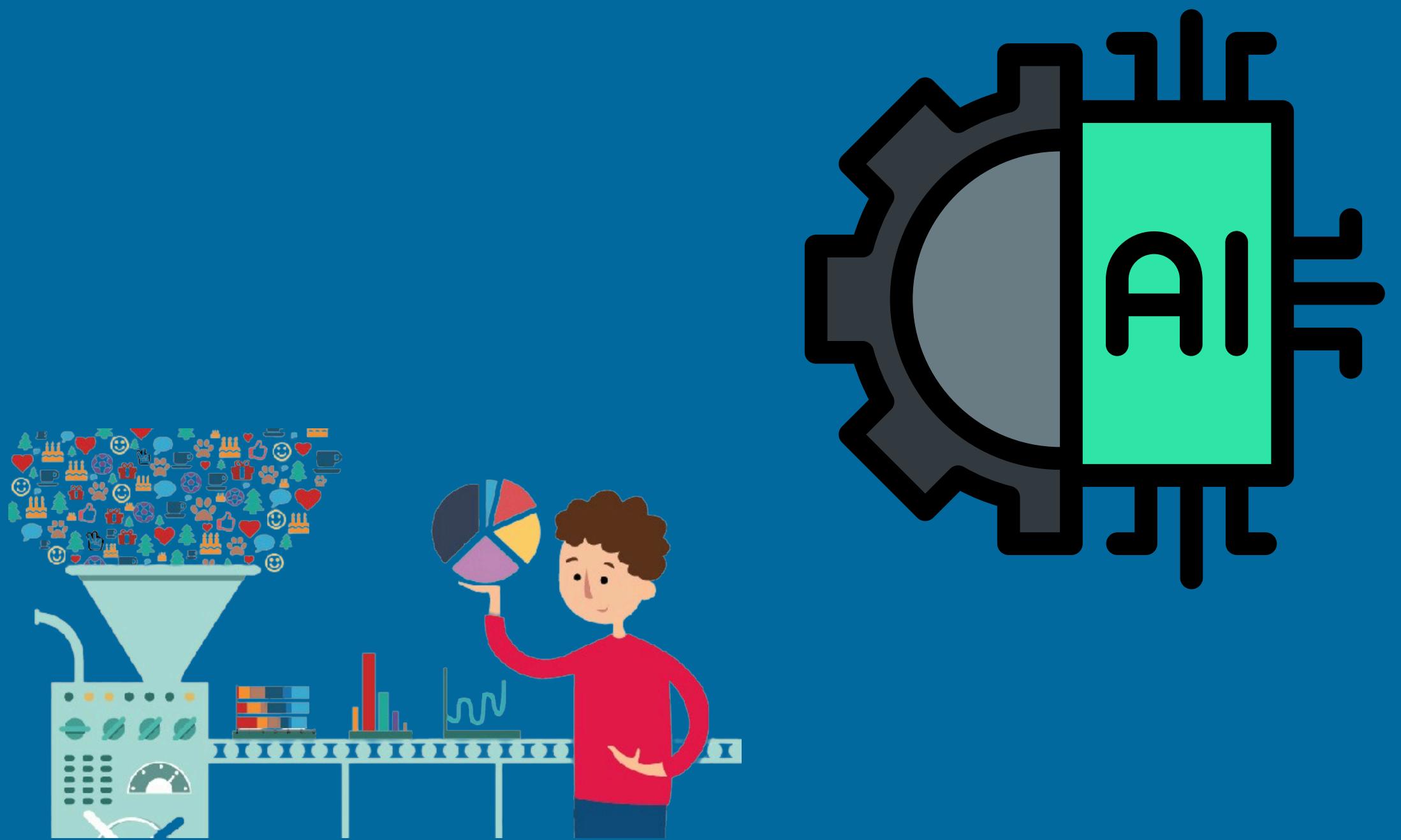
AI MODEL



ໂມເດລທີ່ພວກເຮາໃຊ້ຄົວ **KNN MODEL**



ເຈາລືກ AI MODEL ທີ່ໃຊ້



KNN

MODEL KNN



คือ K-Nearest Neighbors (KNN) เป็นอัลกอริทึมการเรียนรู้แบบมีผู้ดูแลที่ใช้ความใกล้ชิดเพื่อจัดประเภทหรือทำนายการจัดกลุ่มของจุดข้อมูลใหม่

อัลกอริทึมนี้ทำงาน ดังนี้

1. เลือกค่า K: เลือกจำนวน "เพื่อนบ้าน" ที่ใกล้ที่สุดที่ต้องการพิจารณา
2. คำนวณระยะทาง: คำนวณระยะทางระหว่างจุดข้อมูลใหม่และจุดข้อมูลทั้งหมดในชุดข้อมูลการฝึกอบรม
3. ระบุเพื่อนบ้านที่ใกล้ที่สุด: ระบุจุดข้อมูล K ใกล้ที่สุดจากขั้นตอนที่ 2
4. ทำนาย: ทำนายคลาสหรือค่าของจุดข้อมูลใหม่ตามคลาสหรือค่าเฉลี่ยของเพื่อนบ้านที่ใกล้ที่สุด

ทำไมถึงเลือกใช้ KNN MODEL

โมเดล KNN (K-Nearest Neighbor) เป็นหนึ่งในอัลกอริทึมการเรียนรู้ของเครื่องที่มีประสิทธิภาพและใช้งานง่าย
เหมาะอย่างยิ่งสำหรับการพยากรณ์ค่า PM2.5

เหตุผลหลักๆ ที่เลือกใช้ KNN

- ความแม่นยำ:** KNN พิสูจน์แล้วว่ามีประสิทธิภาพสูงในการพยากรณ์ค่า PM2.5 ผลงานวิจัยหลายชิ้นเปรียบเทียบโมเดล KNN กับโมเดลอื่นๆ พบร่วมกันว่า KNN มักให้ผลลัพธ์ที่แม่นยำกว่า
- ความยืดหยุ่น:** KNN สามารถปรับให้เข้ากับข้อมูลประเภทต่างๆ รองรับทั้งข้อมูลเชิงตัวเลข ข้อความ และรูปภาพ โมเดลนี้ไม่มีสมมติฐานที่ซับซ้อน ทำงานได้ดีกับข้อมูลที่ไม่เป็นเส้นตรง
- ความง่ายต่อการใช้งาน:** KNN มีหลักการทำงานที่เข้าใจง่าย ไม่จำเป็นต้องมีความรู้ด้านสถิติหรือคอมพิวเตอร์เซิงลีก เหมาะสำหรับนักวิจัย นักวิเคราะห์ข้อมูล และนักพัฒนาซอฟต์แวร์ทุกระดับ
- ประสิทธิภาพ:** KNN ทำงานได้รวดเร็ว เหมาะสำหรับการวิเคราะห์ข้อมูลขนาดใหญ่
- การเรียนรู้จากข้อมูลใหม่:** KNN สามารถเรียนรู้จากข้อมูลใหม่ได้อย่างต่อเนื่อง โมเดลจะปรับปรุงประสิทธิภาพการพยากรณ์เมื่อมีข้อมูลใหม่อัปเดต

MODEL KNN



ข้อดี

ง่ายต่อการเข้าใจและใช้งาน

มีประสิทธิภาพในการคำนวณ

ทำงานได้ดีกับข้อมูลประเภทต่างๆ

ทำงานกับ Time Series ได้ดี

สามารถใช้สำหรับทั้งงานจัดประเภทและ
การคัดถอย

ข้อเสีย

ประสิทธิภาพอาจช้าลงเมื่อข้อมูลมีขนาดใหญ่

ประสิทธิภาพอาจได้รับผลกระทบจากมิติข้อมูลสูง

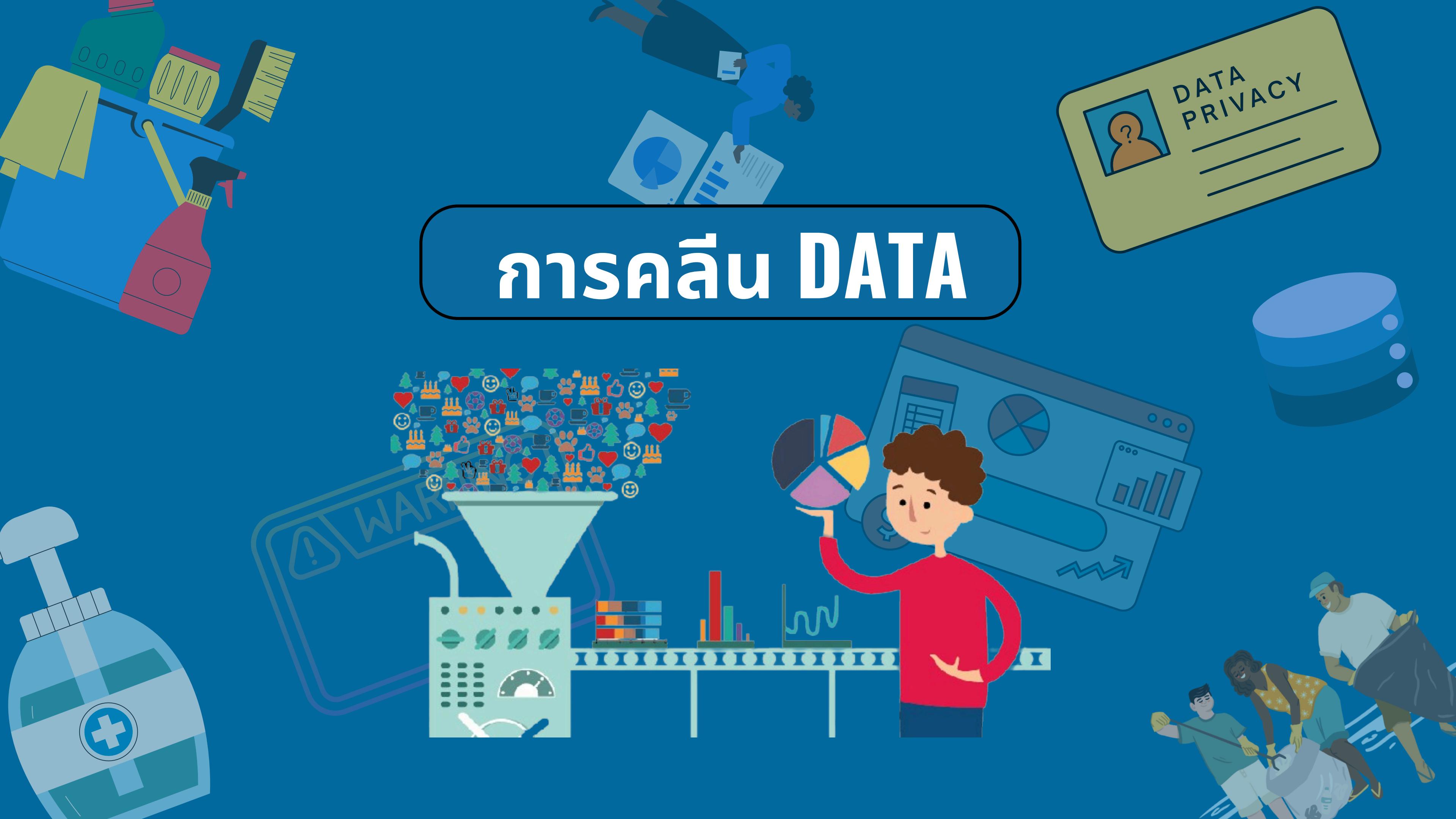
การเลือกค่า K ที่เหมาะสมอาจส่งผลต่อประสิทธิภาพของโมเดล

วิธีการวัดระยะทางที่ใช้ส่งผลต่อผลลัพธ์ของโมเดล ตัวอย่างเช่น ระยะทางแบบ Euclidean อาจไม่เหมาะสมกับข้อมูลที่มีคุณลักษณะบางประเภท

KNN เป็นโมเดลแบบกล่องดำ ยากต่อการตีความว่าโมเดลตัดสินใจอย่างไร



การคลีน DATA



หลังจาก Export DATA มาจาก DASHBOARD ZETA ด้วย CSV แล้ว

จะสังเกตุว่า Data ยังไม่ถูก Clean และแยกส่วนกัน รวมถึงมีหลายส่วนที่ไม่ได้ใช้

A	B	C	D	E	F	G	H	I	J	K	L
1	Data type	Asset number	Asset name	System	Install location	Content	Device type	Device ID	Project	Report time	
2	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:70.5%;PM2.5:95ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:17:04	
3	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:64.5%;PM2.5:79ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:17:01	
4	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	I1-0002-0004-0005-0007, Nun	485 Metering Module	4f0288b2	SPU	2024-04-19 17:16:31	
5	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:70.1%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:16:08	
6	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:64.3%;PM2.5:79ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:16:02	
7	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:69.7%;PM2.5:94ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:15:06	
8	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:63.9%;PM2.5:80ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:15:02	
9	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:69.1%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:14:09	
10	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:63.3%;PM2.5:82ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:14:03	
11	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:68.8%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:13:05	
12	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:63.1%;PM2.5:78ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:13:03	
13	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:68.4%;PM2.5:94ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:12:08	
14	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:62.7%;PM2.5:81ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:12:03	
15	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:67.9%;PM2.5:94ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:11:07	
16	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:62.2%;PM2.5:76ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:11:02	
17	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:67.5%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:10:06	
18	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:62.0%;PM2.5:77ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:10:03	
19	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	idity:67.9%;PM2.5:89ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:09:05	
20	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:62.2%;PM2.5:76ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:09:03	
21	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	I1-0002-0004-0005-0007, Nun	485 Metering Module	4f02889a	SPU	2024-04-19 17:08:45	
22	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	ug/m3;PM10:93ug/m3;CO2:50	485 Metering Module	4f02889a	SPU	2024-04-19 17:08:10	
23	Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	idity:63.1%;PM2.5:75ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:08:03	
24	Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	ug/m3;PM10:97ug/m3;CO2:50	485 Metering Module	4f02889a	SPU	2024-04-19 17:07:10	

ขั้นแรก : เราจะทำการลบในส่วนที่เราไม่ได้ใช้ทิ้งไป ประกอบไปด้วย Data type, Asset number, Asset name, System, Install location, Device type, Device ID, Project

และนี่คือหลังจากได้ลบหั้งคอลัมน์ ที่กล่าวไป

ก่อนลบ

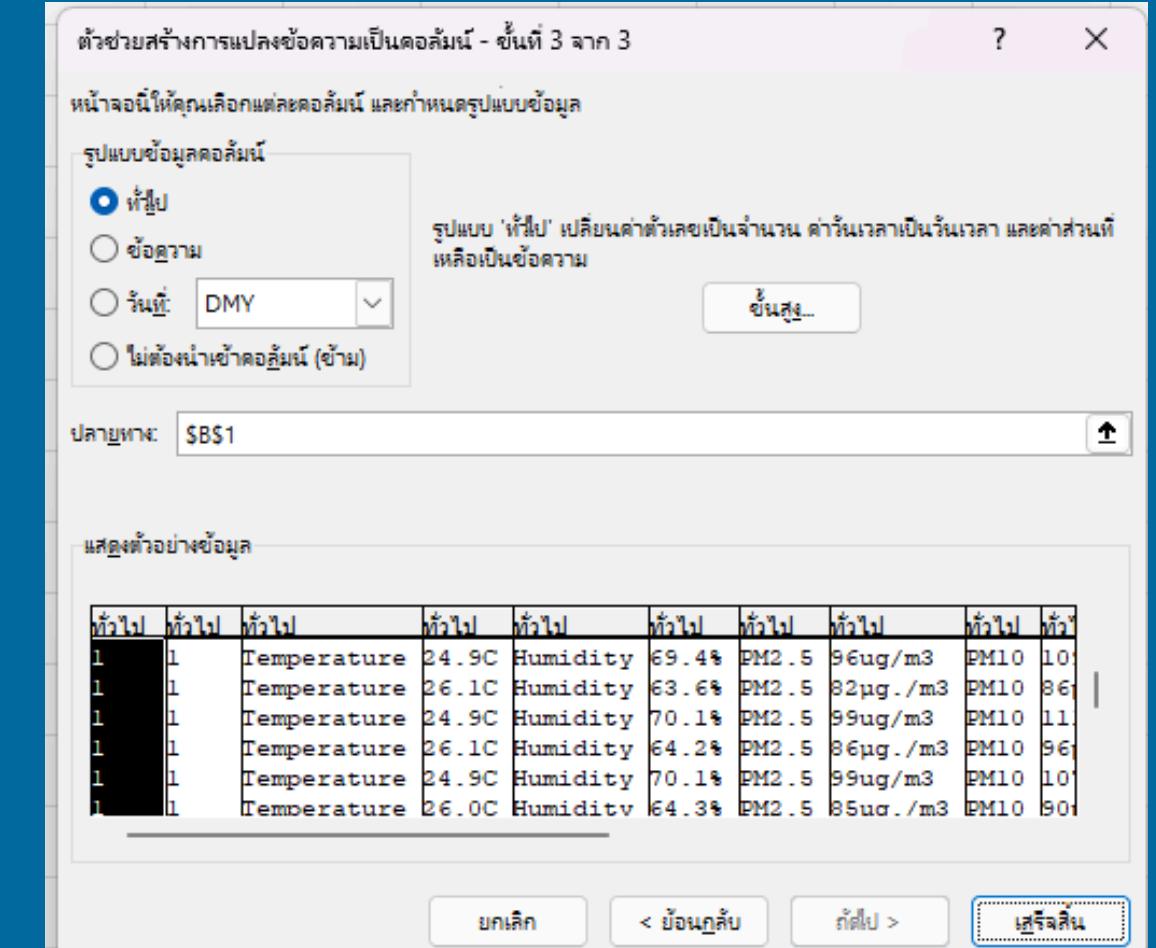
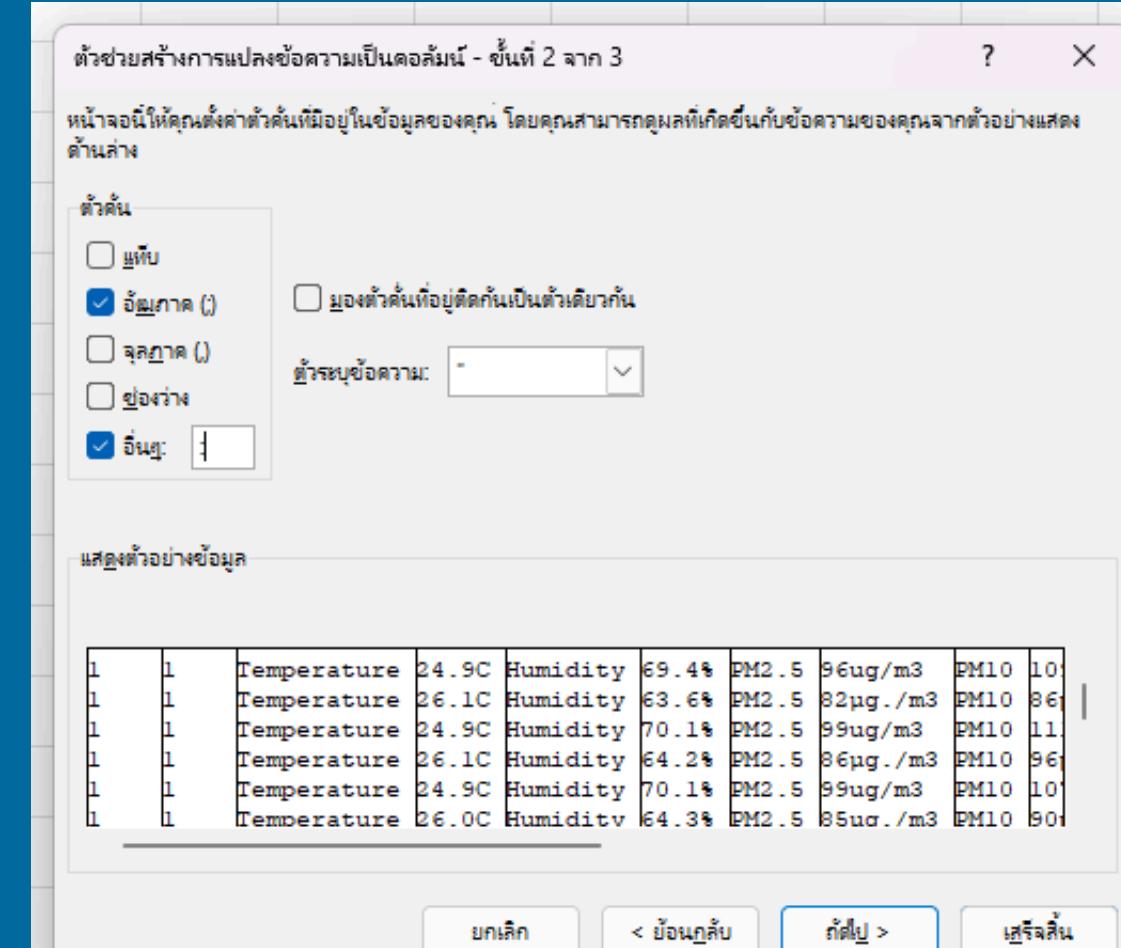
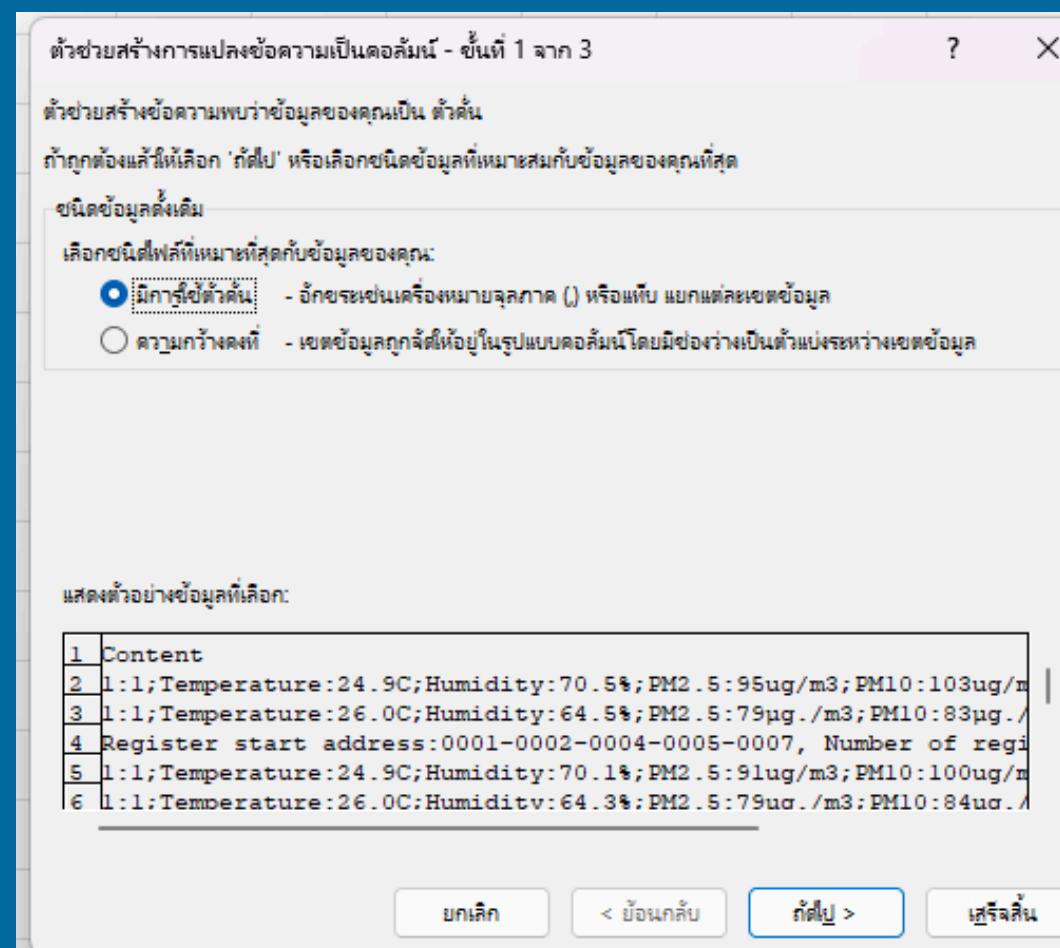
A	B	C	D	E	F	G	H	I	J	K	L
Data type	Asset number	Asset name	System	Install location	Content	Device type	Device ID	Project	Report time		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:70.5%;PM2.5:95ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:17:04		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:64.5%;PM2.5:79ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:17:01		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	1-0002-0004-0005-0007, Num	485 Metering Module	4f0288b2	SPU	2024-04-19 17:16:31		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:70.1%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:16:08		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:64.3%;PM2.5:79ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:16:02		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:69.7%;PM2.5:94ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:15:06		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:63.9%;PM2.5:80ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:15:02		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:69.1%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:14:09		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:63.3%;PM2.5:82ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:14:03		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:68.8%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:13:05		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:63.1%;PM2.5:78ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:13:03		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:68.4%;PM2.5:94ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:12:08		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:62.7%;PM2.5:81ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:12:03		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:67.9%;PM2.5:94ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:11:07		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:62.2%;PM2.5:76ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:11:02		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:67.5%;PM2.5:91ug/m3;P1	485 Metering Module	4f02889a	SPU	2024-04-19 17:10:06		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:62.0%;PM2.5:77ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:10:03		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	dity:67.9%;PM2.5:89ug/m3;P	485 Metering Module	4f02889a	SPU	2024-04-19 17:09:05		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:62.2%;PM2.5:76ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:09:03		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	1-0002-0004-0005-0007, Num	485 Metering Module	4f02889a	SPU	2024-04-19 17:08:45		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	ug/m3;PM10:93ug/m3;CO2:5	485 Metering Module	4f02889a	SPU	2024-04-19 17:08:10		
Normal	SPU-IAQ1	SPU-IAQ1	Others	Area1	dity:63.1%;PM2.5:75ug./m3;P	485 Metering Module	4f0288b2	SPU	2024-04-19 17:08:03		
Normal	SPU-IAQ2	SPU-IAQ2	Others	Area2	ug/m3;PM10:97ug/m3;CO2:5	485 Metering Module	4f02889a	SPU	2024-04-19 17:07:10		

หลังลบ

A	B	C	D	E	F	G
Report time	Content					
2024-04-19 17:17:04	dity:70.5%;PM2.5:95ug/m3;PM10:103ug/m3;CO2:5000PPM					
2024-04-19 17:17:01	dity:64.5%;PM2.5:79ug./m3;PM10:83ug./m3;CO2:2766PPM					
2024-04-19 17:16:31	1-0002-0004-0005-0007, Number of registers:1-1-1-1					
2024-04-19 17:16:08	dity:70.1%;PM2.5:91ug/m3;PM10:100ug/m3;CO2:5000PPM					
2024-04-19 17:16:02	dity:64.3%;PM2.5:79ug./m3;PM10:84ug./m3;CO2:2766PPM					
2024-04-19 17:15:06	dity:69.7%;PM2.5:94ug/m3;PM10:101ug/m3;CO2:5000PPM					
2024-04-19 17:15:02	dity:63.9%;PM2.5:80ug./m3;PM10:83ug./m3;CO2:2759PPM					
2024-04-19 17:14:09	dity:69.1%;PM2.5:91ug/m3;PM10:100ug/m3;CO2:5000PPM					
2024-04-19 17:14:03	dity:63.3%;PM2.5:82ug./m3;PM10:86ug./m3;CO2:2757PPM					
2024-04-19 17:13:05	dity:68.8%;PM2.5:91ug/m3;PM10:102ug/m3;CO2:5000PPM					
2024-04-19 17:13:03	dity:63.1%;PM2.5:78ug./m3;PM10:84ug./m3;CO2:2757PPM					
2024-04-19 17:12:08	dity:68.4%;PM2.5:94ug/m3;PM10:106ug/m3;CO2:5000PPM					
2024-04-19 17:12:03	dity:62.7%;PM2.5:81ug./m3;PM10:88ug./m3;CO2:2775PPM					
2024-04-19 17:11:07	dity:67.9%;PM2.5:94ug/m3;PM10:102ug/m3;CO2:5000PPM					
2024-04-19 17:11:02	dity:62.2%;PM2.5:76ug./m3;PM10:83ug./m3;CO2:2783PPM					
2024-04-19 17:10:06	dity:67.5%;PM2.5:91ug/m3;PM10:103ug/m3;CO2:5000PPM					
2024-04-19 17:10:03	dity:62.0%;PM2.5:77ug./m3;PM10:82ug./m3;CO2:2792PPM					
2024-04-19 17:09:05	dity:67.9%;PM2.5:89ug/m3;PM10:97ug/m3;CO2:5000PPM					
2024-04-19 17:09:03	dity:62.2%;PM2.5:76ug./m3;PM10:81ug./m3;CO2:2784PPM					
2024-04-19 17:08:45	1-0002-0004-0005-0007, Number of registers:1-1-1-1					
2024-04-19 17:08:10	ug/m3;PM10:93ug/m3;CO2:5000PPM;acquisition parameters6:acquisition fail					
2024-04-19 17:08:03	dity:63.1%;PM2.5:75ug./m3;PM10:80ug./m3;CO2:2788PPM					
2024-04-19 17:07:10	ug/m3;PM10:97ug/m3;CO2:5000PPM;acquisition parameters6:acquisition fail					

ขั้นสอง : เราจะทำการแยกแยะใน Content โดยใช้ Text to Column

โดยจะใช้ตัวคัน, ขั้น1 : มีการใช้ตัวคัน, ขั้น2 : เลือก ; และ : , ขั้น3 : เลือก เสร็จสิ้น



ผลลัพธ์ที่ได้หลังแยก data ใน Content

ก่อนแยก

A	B	C	D	E	F	G
Report time	Content					
2024-04-19 17:17:04	dity:70.5%;PM2.5:95ug/m3;PM10:103ug/m3;CO2:5000PPM					
2024-04-19 17:17:01	dity:64.5%;PM2.5:79ug./m3;PM10:83ug./m3;CO2:2766PPm					
2024-04-19 17:16:31	I-0002-0004-0005-0007, Number of registers:1-1-1-1-1					
2024-04-19 17:16:08	dity:70.1%;PM2.5:91ug/m3;PM10:100ug/m3;CO2:5000PPM					
2024-04-19 17:16:02	dity:64.3%;PM2.5:79ug./m3;PM10:84ug./m3;CO2:2766PPm					
2024-04-19 17:15:06	dity:69.7%;PM2.5:94ug/m3;PM10:101ug/m3;CO2:5000PPM					
2024-04-19 17:15:02	dity:63.9%;PM2.5:80ug./m3;PM10:83ug./m3;CO2:2759PPm					
2024-04-19 17:14:09	dity:69.1%;PM2.5:91ug/m3;PM10:100ug/m3;CO2:5000PPM					
2024-04-19 17:14:03	dity:63.3%;PM2.5:82ug./m3;PM10:86ug./m3;CO2:2757PPm					
2024-04-19 17:13:05	dity:68.8%;PM2.5:91ug/m3;PM10:102ug/m3;CO2:5000PPM					
2024-04-19 17:13:03	dity:63.1%;PM2.5:78ug./m3;PM10:84ug./m3;CO2:2757PPm					
2024-04-19 17:12:08	dity:68.4%;PM2.5:94ug/m3;PM10:106ug/m3;CO2:5000PPM					
2024-04-19 17:12:03	dity:62.7%;PM2.5:81ug./m3;PM10:88ug./m3;CO2:2775PPm					
2024-04-19 17:11:07	dity:67.9%;PM2.5:94ug/m3;PM10:102ug/m3;CO2:5000PPM					
2024-04-19 17:11:02	dity:62.2%;PM2.5:76ug./m3;PM10:83ug./m3;CO2:2783PPm					
2024-04-19 17:10:06	dity:67.5%;PM2.5:91ug/m3;PM10:103ug/m3;CO2:5000PPM					
2024-04-19 17:10:03	dity:62.0%;PM2.5:77ug./m3;PM10:82ug./m3;CO2:2792PPm					
2024-04-19 17:09:05	I-dity:67.9%;PM2.5:89ug/m3;PM10:97ug/m3;CO2:5000PPM					
2024-04-19 17:09:03	dity:62.2%;PM2.5:76ug./m3;PM10:81ug./m3;CO2:2784PPm					
2024-04-19 17:08:45	I-0002-0004-0005-0007, Number of registers:1-1-1-1-1					
2024-04-19 17:08:10	ug/m3;PM10:93ug/m3;CO2:5000PPM;acquisition parameters6:acquisition fail					
2024-04-19 17:08:03	dity:63.1%;PM2.5:75ug./m3;PM10:80ug./m3;CO2:2788PPm					
2024-04-19 17:07:10	ug/m3;PM10:97ug/m3;CO2:5000PPM;acquisition parameters6:acquisition fail					

หลังแยก

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Report time	Content														
2024-04-19 17:17:04	1	1 Temperati 24.9C	Humidity	70.50%	PM2.5	95ug/m3	PM10	103ug/m3	CO2	5000PPM					
2024-04-19 17:17:01	1	1 Temperati 26.0C	Humidity	64.50%	PM2.5	79ug./m3	PM10	83ug./m3	CO2	2766PPm					
2024-04-19 17:16:31	Register start address 0001-0002 1-1-1-1-1	1 Temperati 24.9C	Humidity	70.10%	PM2.5	91ug/m3	PM10	100ug/m3	CO2	5000PPM					
2024-04-19 17:16:08	1	1 Temperati 26.0C	Humidity	64.30%	PM2.5	79ug./m3	PM10	84ug./m3	CO2	2766PPm					
2024-04-19 17:16:02	1	1 Temperati 24.9C	Humidity	69.70%	PM2.5	94ug/m3	PM10	101ug/m3	CO2	5000PPM					
2024-04-19 17:15:06	1	1 Temperati 26.0C	Humidity	63.90%	PM2.5	80ug./m3	PM10	83ug./m3	CO2	2759PPm					
2024-04-19 17:15:02	1	1 Temperati 24.9C	Humidity	69.10%	PM2.5	91ug/m3	PM10	100ug/m3	CO2	5000PPM					
2024-04-19 17:14:09	1	1 Temperati 26.0C	Humidity	63.30%	PM2.5	82ug./m3	PM10	86ug./m3	CO2	2757PPm					
2024-04-19 17:14:03	1	1 Temperati 24.9C	Humidity	68.80%	PM2.5	91ug/m3	PM10	102ug/m3	CO2	5000PPM					
2024-04-19 17:13:05	1	1 Temperati 26.0C	Humidity	63.10%	PM2.5	78ug./m3	PM10	84ug./m3	CO2	2757PPm					
2024-04-19 17:13:03	1	1 Temperati 24.9C	Humidity	68.40%	PM2.5	94ug/m3	PM10	106ug/m3	CO2	5000PPM					
2024-04-19 17:12:08	1	1 Temperati 26.0C	Humidity	62.70%	PM2.5	81ug./m3	PM10	88ug./m3	CO2	2775PPm					
2024-04-19 17:12:03	1	1 Temperati 24.9C	Humidity	67.90%	PM2.5	94ug/m3	PM10	102ug/m3	CO2	5000PPM					
2024-04-19 17:11:07	1	1 Temperati 26.0C	Humidity	62.20%	PM2.5	76ug./m3	PM10	83ug./m3	CO2	2783PPm					
2024-04-19 17:11:02	1	1 Temperati 24.9C	Humidity	67.50%	PM2.5	91ug/m3	PM10	103ug/m3	CO2	5000PPM					
2024-04-19 17:10:06	1	1 Temperati 26.1C	Humidity	62.00%	PM2.5	77ug./m3	PM10	82ug./m3	CO2	2792PPm					
2024-04-19 17:10:03	1	1 Temperati 24.9C	Humidity	67.90%	PM2.5	89ug/m3	PM10	97ug/m3	CO2	5000PPM					
2024-04-19 17:09:05	1	1 Temperati 26.1C	Humidity	62.20%	PM2.5	76ug./m3	PM10	81ug./m3	CO2	2784PPm					
2024-04-19 17:09:03	Register start address 0001-0002 1-1-1-1-1	1 Temperati 24.9C	Humidity	68.80%	PM2.5	89ug/m3	PM10	93ug/m3	CO2	5000PPM	acquisitio	acquisition	fail		
2024-04-19 17:08:45	1	1 Temperati 26.1C	Humidity	63.10%	PM2.5	75ug./m3	PM10	80ug./m3	CO2	2788PPm					
2024-04-19 17:08:10	1	1 Temperati 24.9C	Humidity	69.70%	PM2.5	89ug/m3	PM10	97ug/m3	CO2	5000PPM	acquisitio	acquisition	fail		
2024-04-19 17:08:03	1	1 Temperati 24.9C	Humidity	63.30%	PM2.5	91ug/m3	PM10	103ug/m3	CO2	5000PPM	acquisitio	acquisition	fail		
2024-04-19 17:07:10	1	1 Temperati 24.9C	Humidity	67.50%	PM2.5	89ug/m3	PM10	97ug/m3	CO2	5000PPM	acquisitio	acquisition	fail		

**ขึ้นสาม : ทำการลบในส่วนของ ที่เกินมาและไม่ได้ใช้ หลังจากนั้น นำมารัดใหม่ให้สวยงาม
พร้อมใช้**

หลังเคลียร์และ จัดการแล้ว

A	B	C	D	E	F
date_time	Temperature	Humidity	PM2.5	PM10	CO2
2024-04-19 17:17:04	24.9C	70.50%	95ug/m3	103ug/m3	5000PPM
2024-04-19 17:17:01	26.0C	64.50%	79ug./m3	83ug./m3	2766PPm
2024-04-19 17:16:31					
2024-04-19 17:16:08	24.9C	70.10%	91ug/m3	100ug/m3	5000PPM
2024-04-19 17:16:02	26.0C	64.30%	79ug./m3	84ug./m3	2766PPm
2024-04-19 17:15:06	24.9C	69.70%	94ug/m3	101ug/m3	5000PPM
2024-04-19 17:15:02	26.0C	63.90%	80ug./m3	83ug./m3	2759PPm
2024-04-19 17:14:09	24.9C	69.10%	91ug/m3	100ug/m3	5000PPM
2024-04-19 17:14:03	26.0C	63.30%	82ug./m3	86ug./m3	2757PPm
2024-04-19 17:13:05	24.9C	68.80%	91ug/m3	102ug/m3	5000PPM
2024-04-19 17:13:03	26.0C	63.10%	78ug./m3	84ug./m3	2757PPm
2024-04-19 17:12:08	24.9C	68.40%	94ug/m3	106ug/m3	5000PPM
2024-04-19 17:12:03	26.0C	62.70%	81ug./m3	88ug./m3	2775PPm
2024-04-19 17:11:07	24.9C	67.90%	94ug/m3	102ug/m3	5000PPM
2024-04-19 17:11:02	26.1C	62.20%	76ug./m3	83ug./m3	2783PPm
2024-04-19 17:10:06	24.9C	67.50%	91ug/m3	103ug/m3	5000PPM
2024-04-19 17:10:03	26.1C	62.00%	77ug./m3	82ug./m3	2792PPm
2024-04-19 17:09:05	24.9C	67.90%	89ug/m3	97ug/m3	5000PPM
2024-04-19 17:09:03	26.1C	62.20%	76ug./m3	81ug./m3	2784PPm
2024-04-19 17:08:45					

ขั้นสุดท้าย : ทำการเคลียร์หนวยในช่องทุกตัวเนื่องจากว่า ตอนจะนำเข้าไมเดลนั้นจำเป็นต้องไม่มีหนวยอยู่ในช่อง

A	B	C	D	E	F
date_time	Temperature	Humidity	PM2.5	PM10	CO2
2024-04-19 17:17:04	24.90	0.71	95.00	103.00	5000.00
2024-04-19 17:17:01	26.00	0.65	79.00	83.00	2766.00
2024-04-19 17:16:31					
2024-04-19 17:16:08	24.90	0.70	91.00	100.00	5000.00
2024-04-19 17:16:02	26.00	0.64	79.00	84.00	2766.00
2024-04-19 17:15:06	24.90	0.70	94.00	101.00	5000.00
2024-04-19 17:15:02	26.00	0.64	80.00	83.00	2759.00
2024-04-19 17:14:09	24.90	0.69	91.00	100.00	5000.00
2024-04-19 17:14:03	26.00	0.63	82.00	86.00	2757.00
2024-04-19 17:13:05	24.90	0.69	91.00	102.00	5000.00
2024-04-19 17:13:03	26.00	0.63	78.00	84.00	2757.00
2024-04-19 17:12:08	24.90	0.68	94.00	106.00	5000.00
2024-04-19 17:12:03	26.00	0.63	81.00	88.00	2775.00
2024-04-19 17:11:07	24.90	0.68	94.00	102.00	5000.00
2024-04-19 17:11:02	26.10	0.62	76.00	83.00	2783.00
2024-04-19 17:10:06	24.90	0.68	91.00	103.00	5000.00
2024-04-19 17:10:03	26.10	0.62	77.00	82.00	2792.00
2024-04-19 17:09:05	24.90	0.68	89.00	97.00	5000.00
2024-04-19 17:09:03	26.10	0.62	76.00	81.00	2784.00
2024-04-19 17:08:45					

**หลังเคลียร์แล้ว
จัดการแล้ว**

หลังจากนั้นจะทำการ Export เป็น CSV แล้ว นำเข้าไมเดล 

หลังเคลียร์แล้ว
จัดการแล้ว

	A	B	C	D	E	F
1	date_time	Temperature	Humidity	PM2.5	PM10	CO2
2	2024-04-19 17:17:04	24.90	0.71	95.00	103.00	5000.00
3	2024-04-19 17:17:01	26.00	0.65	79.00	83.00	2766.00
4	2024-04-19 17:16:31					
5	2024-04-19 17:16:08	24.90	0.70	91.00	100.00	5000.00
6	2024-04-19 17:16:02	26.00	0.64	79.00	84.00	2766.00
7	2024-04-19 17:15:06	24.90	0.70	94.00	101.00	5000.00
8	2024-04-19 17:15:02	26.00	0.64	80.00	83.00	2759.00
9	2024-04-19 17:14:09	24.90	0.69	91.00	100.00	5000.00
10	2024-04-19 17:14:03	26.00	0.63	82.00	86.00	2757.00
11	2024-04-19 17:13:05	24.90	0.69	91.00	102.00	5000.00
12	2024-04-19 17:13:03	26.00	0.63	78.00	84.00	2757.00
13	2024-04-19 17:12:08	24.90	0.68	94.00	106.00	5000.00
14	2024-04-19 17:12:03	26.00	0.63	81.00	88.00	2775.00
15	2024-04-19 17:11:07	24.90	0.68	94.00	102.00	5000.00
16	2024-04-19 17:11:02	26.10	0.62	76.00	83.00	2783.00
17	2024-04-19 17:10:06	24.90	0.68	91.00	103.00	5000.00
18	2024-04-19 17:10:03	26.10	0.62	77.00	82.00	2792.00
19	2024-04-19 17:09:05	24.90	0.68	89.00	97.00	5000.00
20	2024-04-19 17:09:03	26.10	0.62	76.00	81.00	2784.00
	2024-04-19 17:08:45					

หลังจากนั้นจะทำการ Export เป็น CSV แล้ว นำเข้าโนําเดา ➔

การนำข้อมูลที่ได้หลังคลีบ ไปเข้าโมเดล **KNN**



ภาษาและโปรแกรมที่จะนำมาใช้คือ

PYTHON



Google colaboratory



Java

MYSQL

JS

PHP

เราจะทำการติดตั้ง Library ที่จะนำไปใช้เข้ามาคือ Panda, scikit-learn, matplotlib, seaborn

The screenshot shows a Jupyter Notebook interface with two code cells. The top cell contains the command `!pip install pandas scikit-learn matplotlib`. The output of this cell lists numerous dependencies that are already satisfied, including pandas, scikit-learn, matplotlib, and various sub-dependencies like numpy, scipy, and pillow. The bottom cell contains the command `pip install seaborn`, which also lists its dependencies that are already satisfied, including seaborn, numpy, pandas, and matplotlib.

```
495-KNN-SUCESS.ipynb ☆
ไฟล์ แก้ไข บุคลิก แทรก รันไทม์ เครื่องมือ ความช่วยเหลือ
แสดงความคิดเห็น แชร์ ⚙️
+ โค้ด + ข้อความ RAM ดีล็อก 🔍 ⌂ ⌃ ⌄
495 KNN-SUCESS
{x} 5 วินาที
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.0.3)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (3.7.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.1)
Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/python3.10/dist-packages (from pandas) (1.25.2)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.4.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4.5)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (24.0)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.1.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)

pip install seaborn
Requirement already satisfied: seaborn in /usr/local/lib/python3.10/dist-packages (0.13.1)
Requirement already satisfied: numpy!=1.24.0,>=1.20 in /usr/local/lib/python3.10/dist-packages (from seaborn) (1.25.2)
Requirement already satisfied: pandas>=1.2 in /usr/local/lib/python3.10/dist-packages (from seaborn) (2.0.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in /usr/local/lib/python3.10/dist-packages (from seaborn) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (1.4.5)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (24.0)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.2->seaborn) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.2->seaborn) (2024.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.4->seaborn) (1.16.0)
```

1. จากนั้นจะทำการ Upload data จากข้อมูลในเครื่องเข้ามา

```
▶ from google.colab import files  
  
# Upload data.csv file  
uploaded = files.upload()  
  
↳ [เลือกไฟล์] Hisdata1-1.csv  
• Hisdata1-1.csv(text/csv) - 3253119 bytes, last modified: 21/4/2567 - 100% done  
Saving Hisdata1-1.csv to Hisdata1-1.csv
```

2. และจากนั้นจะทำการimport library ที่เราติดตั้งมาใช้

```
▶ import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
from sklearn.model_selection import train_test_split  
from sklearn.neighbors import KNeighborsRegressor  
from sklearn.metrics import mean_squared_error
```

3. จะเริ่มด้วยการ อ่านข้อมูลจากไฟล์ที่นำเข้ามาพร้อมทั้งให้ช่วง 5 และแทรกเพื่อดูข้อมูลคร่าวๆ

```
[ ] # 1. อ่านข้อมูลจาก CSV file  
file_path = 'Hisdata1-1.csv'  
df = pd.read_csv(file_path)  
df.head()
```

	date_time	Temperature	Humidity	PM2.5	PM10	co2
0	19/4/2024 17:17	24.9	0.71	95.0	103.0	5000.0
1	19/4/2024 17:17	26	0.65	79.0	83.0	2766.0
2	19/4/2024 17:16	Nan	Nan	Nan	Nan	Nan
3	19/4/2024 17:16	24.9	0.70	91.0	100.0	5000.0
4	19/4/2024 17:16	26	0.64	79.0	84.0	2766.0

4. จากนี้เมื่อทำการ Check missing value แล้วจะพบว่ามี missing value ก็งหมดถึง 942 ตัว

```
▶ print(df.isnull().sum()) # Check for missing values  
  
date_time      0  
Temperature    942  
Humidity      942  
PM2.5         942  
PM10          942  
CO2           942  
dtype: int64
```

6. แล้วจะทำการเช็ค 5 แถวแรกใหม่เพื่อดูว่า missing value หมดหรือยัง

```
▶ df.head()  
  
date_time  Temperature  Humidity  PM2.5  PM10  CO2  
0  19/4/2024 17:17      24.9     0.71   95.0  103.0  5000.0  
1  19/4/2024 17:17      26.0     0.65   79.0   83.0  2766.0  
3  19/4/2024 17:16      24.9     0.70   91.0  100.0  5000.0  
4  19/4/2024 17:16      26.0     0.64   79.0   84.0  2766.0  
5  19/4/2024 17:15      24.9     0.70   94.0  101.0  5000.0  
  
Next steps: View recommended plots
```

5. หลังจากลบแล้วจะทำการกำจัด missing value และเช็คว่า missing value หมดยัง

```
▶ df.dropna(inplace=True) #missing value เพื่อฯ  
print(df.isnull().sum()) # Check for missing values  
  
date_time      0  
Temperature    0  
Humidity      0  
PM2.5         0  
PM10          0  
CO2           0  
dtype: int64
```

7. จากนี้เราจะทำการเช็คแล้วก็จะพบว่ามีกี่คอลัมน์จะพบว่ามีก็ 88678 คอลัมน์

```
▶ df.info()  
  
<class 'pandas.core.frame.DataFrame'>  
Index: 88678 entries, 0 to 89619  
Data columns (total 6 columns):  
 #   Column      Non-Null Count Dtype  
---  
 0   date_time   88678 non-null object  
 1   Temperature 88678 non-null object  
 2   Humidity    88678 non-null float64  
 3   PM2.5       88678 non-null float64  
 4   PM10        88678 non-null float64  
 5   CO2         88678 non-null float64  
dtypes: float64(4), object(2)  
memory usage: 6.8+ MB
```

8. ทำการแปลงข้อมูลก่อนเข้าโมเดล

- แปลงจาก date_time เป็น datetime object

```
# แปลงคอลัมน์ date_time เป็น datetime object  
df['date_time'] = pd.to_datetime(df['date_time'], format='%d/%m/%Y %H:%M')  
  
# ตั้งค่า date_time เป็น index ของ DataFrame  
df.set_index('date_time', inplace=True)
```

- เนื่องจากในบางแควมีข้อความหรือข้อมูลที่ไม่
ปถูกตัวแล้วการใช้ dropna ไม่ช่วยจึงทำการกรอง
ข้อมูล และเปลี่ยนค่าชนิด pm2.5 ให้เป็น ตัวเลข

```
# กรองข้อมูลโดยลบแค่ที่มีค่า 'aquisition fail' ในคอลัมน์ 'PM2.5'  
df = df[df['PM2.5'] != 'aquisition fail']  
  
# แปลงค่าในคอลัมน์ 'PM2.5' เป็นชนิดตัวเลข  
df['PM2.5'] = pd.to_numeric(df['PM2.5'], errors='coerce')
```

- ทำการเลือก features ที่เราต้องการ
ใช้ในการทำนาย และ เลือกเป้าหมาย
ที่เราจะต้องการทำนาย

```
# เลือก features ที่ต้องการใช้ในการทำนาย (pm2.5, hcho, tvoc, temperature, humidity)  
features = ['Temperature', 'Humidity', 'PM10', 'CO2']  
  
# แยก features และ target variable (pm2.5)  
X = df[features]  
y = df['PM2.5']
```

9.นำข้อมูลหลังแปลงมาเข้าโมเดล

-ทำการแบ่งชุดข้อมูลเป็น train กับ test

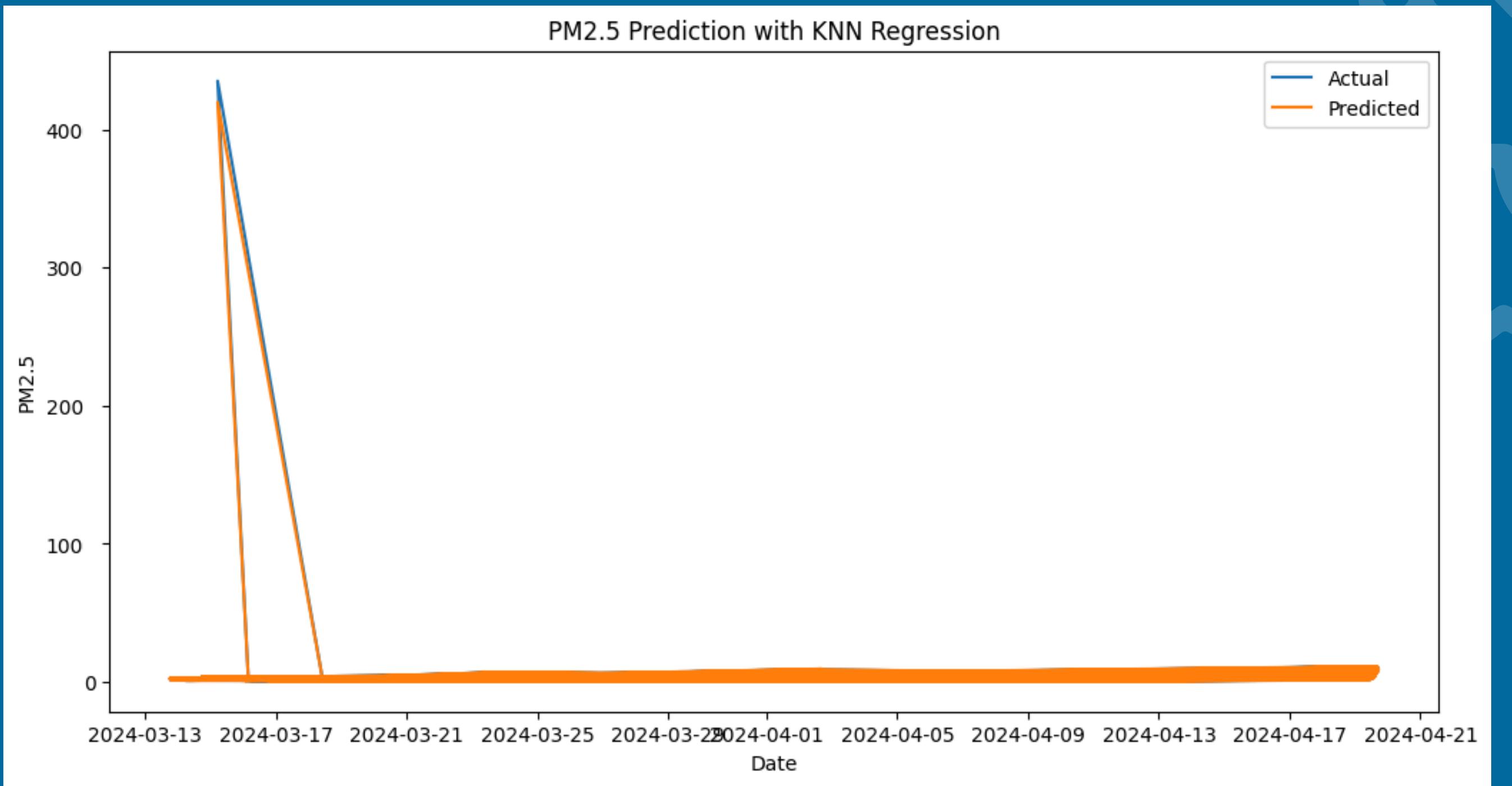
```
▶ # แบ่งข้อมูลเป็นชุด train และ test  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)  
  
# ตรวจสอบและลบแถวที่มีค่า 'aquisition fail' ใน X_train และ y_train  
train_indices = X_train.index[X_train.isin(['aquisition fail']).any(axis=1)]  
X_train = X_train[~X_train.index.isin(train_indices)]  
y_train = y_train[~y_train.index.isin(train_indices)]
```

-สร้างและฝึกโมเดลขึ้นมาโดยทำการกำหนดค่า default ให้จำนวนที่ใช้ในการทำนายคือ 5 และทำการทำนาย

```
▶ # สร้างและฝึกโมเดล KNN Regression  
n_neighbors = 5 # จำนวนของเพื่อนบ้านที่ใช้ในการทำนาย  
knn_model = KNeighborsRegressor(n_neighbors=n_neighbors, weights='uniform', metric='euclidean')  
knn_model.fit(X_train, y_train)  
  
# ทดสอบโมเดลบนชุดข้อมูลทดสอบ  
y_pred = knn_model.predict(X_test)
```

10. ทำการพอร์ตกราฟในการทำนาย

```
# พล็อตกราฟผลลัพธ์การทำนาย
plt.figure(figsize=(12, 6))
plt.plot(y_test.index, y_test, label='Actual')
plt.plot(y_test.index, y_pred, label='Predicted')
plt.title('PM2.5 Prediction with KNN Regression')
plt.xlabel('Date')
plt.ylabel('PM2.5')
plt.legend()
plt.show()
```



จากกราฟ เราสามารถเห็นได้ว่าค่า PM2.5 ที่คาดการณ์นั้นใกล้เคียงกับค่า PM2.5 จริงแต่เมื่อวันที่ค่า PM2.5 ที่คาดการณ์นั้นสูงกว่าค่า PM2.5 จริง ตัวอย่างเช่นประมาณในวันที่ 2024-03-17 ค่า PM2.5 จริงอยู่ที่ประมาณ 200 ไมโครกรัมต่อลูกบาศก์เมตร แต่ค่า PM2.5 ที่คาดการณ์นั้นอยู่ที่ประมาณ 250 ไมโครกรัมต่อลูกบาศก์เมตร

11. ทำการวัดประสิทธิภาพโดยใช้ MSE , RRSE, R-SQUARE

```
[ ] # คำนวณค่า Mean Squared Error (MSE)
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')
```

Mean Squared Error: 2.0997113216057794

```
▶ from sklearn.metrics import r2_score

# คำนวณค่า Mean Squared Error (MSE)
mse = mean_squared_error(y_test, y_pred)

# คำนวณ Root Mean Squared Error (RMSE)
rmse = np.sqrt(mse)

# คำนวณ Root Relative Squared Error (RRSE)
mean_y_test = np.mean(y_test)
rrse = rmse / mean_y_test

# คำนวณค่า R-square (Coefficient of Determination)
r2 = r2_score(y_test, y_pred)

print(f'Root Relative Squared Error (RRSE): {rrse}')
print(f'R-square (Coefficient of Determination): {r2}')
```

→ Root Relative Squared Error (RRSE): 0.06552046207863133
R-square (Coefficient of Determination): 0.9981895082200753

MSE : 2.099

RRSE : 0.065

R-SQUARE : 0.998

Mean Squared Error (MSE)

- MSE หรือ Mean Squared Error เป็นหนึ่งในวิธีการวัดประสิทธิภาพของโมเดลในการพยากรณ์หรือการทำนายข้อมูล โดยคำนวณจากการหาค่าต่าง ๆ ระหว่างค่าที่โมเดลทำนายได้กับค่าจริงของข้อมูลทดสอบ จากนั้นทำการยกกำลังสองของค่าต่าง ๆ นี้และหาค่าเฉลี่ย
- MSE ที่น้อยกว่า 1 มักถือเป็นการทำนายที่ดี เนื่องจากค่า MSE นี้มักจะแสดงให้เห็นถึงความแม่นยำของโมเดลในการทำนายข้อมูล
- MSE ที่อยู่ระหว่าง 1 ถึง 5 อาจถือเป็นค่าที่ยอมรับได้สำหรับสถานการณ์ทั่วไป
- หาก MSE เกิน 5 อาจแสดงถึงความไม่แม่นยำในการทำนายข้อมูล และอาจต้องพิจารณาปรับปรุงโมเดลหรือการใช้ข้อมูลเพิ่มเติม
- ในกรณีที่ค่า $MSE = 2.099$ ถือว่ามีประสิทธิภาพในการทำนายที่ยอมรับได้ แต่ควรพิจารณาเป็นเพียงข้อมูลเพื่อที่จะวิเคราะห์ความสอดคล้องของโมเดล

Root Relative Squared Error (RRSE):

- RRSE เป็นตัวบ่งชี้ประสิทธิภาพของโมเดลในการทำนายค่า $\text{pm}2.5$ โดยวัดความคลาดเคลื่อนระหว่างค่าที่ทำนายได้กับค่าจริงของ $\text{pm}2.5$
- ค่า RRSE ที่น้อยกว่าหมายถึงโมเดลมีความแม่นยำมากขึ้น
- ในกรณีที่ค่า RRSE = 0.065 แสดงว่าโมเดลมีความคลาดเคลื่อนระดับน้อยเมื่อเทียบกับค่าจริง และมีความแม่นยำสูง

Squared Correlation:

- Squared correlation เป็นตัวบ่งชี้ความสัมพันธ์ระหว่างค่าที่ทำนาย และค่าจริงของ $\text{pm}2.5$ โดยวัดความสอดคล้องของข้อมูลที่ทำนายได้กับข้อมูลจริง
- ค่า Squared correlation อยู่ในช่วง 0-1 โดยค่าที่ใกล้เคียง 1 แสดงถึงความสอดคล้องที่สูงของโมเดลในการทำนาย
- ในกรณีที่ Squared correlation = 0.998 แสดงว่าโมเดลมีความสอดคล้องกับข้อมูลจริงอย่างดี

12. ทำการหา correlation เพื่อดูว่า ค่าไหนส่งผลต่อการทำให้เกิด PM2.5 มากที่สุด

```
import seaborn as sns

# รวม X_train และ y_train เพื่อคำนวณ correlation
df_train = pd.concat([X_train, y_train], axis=1)

# คำนวณ correlation matrix
correlation_matrix = df_train.corr()

# พล็อต heatmap ของ correlation matrix
plt.figure(figsize=(10, 8))
heatmap = sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', linewidths=.5)
plt.title('Correlation Heatmap')
plt.show()
```

เมตริกซ์นี้จะสามารถดูได้ว่าค่าไหนส่งผลต่อการทำให้เกิดค่ามากที่สุดโดยถ้าเราโฟกัสที่ PM2.5 จะเห็นว่า PM10 จะมีค่าเป็น 0.81 ซึ่งมีค่ามากที่สุด ซึ่งเป็นตัวตั้งต้นที่ทำให้เกิด PM2.5 มากที่สุด

