Data Journey Rain Fall in Thailand (1987-2022) by Kruawun Jankaew

1. การโหลด data html (htm) เข้ามาใน Pandas เพื่อมาสร้างเป็น dataframe ตอนแรกทำไม่ได้ มี error ขึ้นมาว่าข้อมูลเป็น list object ไม่สามารถแสดง head(), sample(), tail() ได้

   สิ่งที่ทำ: เช็คให้แน่ใจว่าข้อมูลมี table เดียวหรือไม่ แล้วก็พบว่าในข้อมูลที่ได้รับมาเป็น .htm ไฟล์นั้นมีส่วนของหมายเหตุข้างล่าง ซึ่งถูกมองว่าเป็น table ที่ 2 และข้อมูลทั้ง 2 tables จึงถูก Pandas มองเป็น list ดังนั้นจึงได้กำหนดให้ dataframe คือข้อมูลจาก table[0]
2. ทำการ select/access row และ column ที่ต้องการนำมาทำงานต่อ คือ row (index) ที่ 2 เป็นต้นไป (ไม่เอา headers) และ column ที่ 1 เป็นต้นไป (เอาข้อมูลสถานีและข้อมูลน้ำฝน)
3. ชื่อ column name ถูก assign อัตโนมัติให้เป็น 1-15

   สิ่งที่ทำ: ทำการ rename column name จาก 1 เป็น Station, เปลี่ยน 2 เป็น Yr, เปลี่ยน 3 – 14 เป็น Jan – Dec และเปลี่ยน 15 เป็น Ave เพื่อให้สะดวกกับการ check data และเพื่อสะดวกต่อการอ้างอิงถ้าต้อง split หรือ concat
4. ข้อมูลบางค่า เป็น ‘T’, ‘-‘ ซึ่งทำให้ไม่สามารถทำ operation ต่าง ๆได้ เช่น ถ้าอยากหาเดือนที่ฝนตกเยอะที่สุด ทั้งนี้จากการสอบถามไปทางกรมอุตุนิยมวิทยาถึงความหมายของ ‘T’ และ ‘-‘ ก็ได้รับแจ้งว่า ‘T’ คือ trace หรือมีปริมาณน้ำฝนที่วัดได้น้อยมาก น้อยกว่า 0.1 ml ส่วน ‘-‘ ทางกรมอุตุแจ้งว่าคือ ‘ไม่มีฝนตก’ แต่จากการตรวจสอบข้อมูลค่าเฉลี่ยที่รายงานมาในแต่ละสถานีนั้นพบว่าจำนวนเดือนที่เอามาเฉลี่ยไม่นับเดือนที่มี ‘-‘ จึงมั่นใจว่า ‘-‘ แปลว่า ไม่ได้วัด หรือไม่มีข้อมูล

   สิ่งที่ทำ: เช็ค type ของข้อมูลพบว่าเป็น object จึงต้องทำการเปลี่ยน type เป็น string เพื่อให้สามารถเปลี่ยน ‘T’ และ ‘-‘ เป็น 0 และ np.nan หลังจากนั้นจึงทำการเปลี่ยน type ของข้อมูลใน column ที่เป็นเดือนและ Ave จาก String ให้เป็น float เพื่อที่จะสามารถทำ operation อื่น ๆต่อไปได้
5. ต้องการทำการ group ข้อมูลน้ำฝนจาก stations ต่าง ๆใน 1 จังหวัดต่อปีให้เป็นค่าเดียว เพราะตอนนี้ต่อ 1 จังหวัดอาจจะมีข้อมูลหลายสถานี

   สิ่งที่ทำ: ทำการ split column ชื่อ Station เพื่อแยกออกมาเป็น Station number(name) และ Province โดยใช้ ‘ จ.’ เป็นตัว split หลังจากนั้น สร้าง column ใหม่ขึ้นมาชื่อ YrProvince ซึ่งเป็น column ที่ concat ระหว่าง column ชื่อ Yr และ Province หลังจากนั้นสร้าง dataframe ใหม่ขึ้นมาจากการ groupby YrProvince
6. หลังจากพิจารณาข้อมูลทั้งหมดแล้ว ซึ่งตอนแรกยังไม่มั่นใจว่าควรจะแสดงผลข้อมูลน้ำฝนเฉลี่ยรายเดือนต่อจังหวัด โดย 1 เส้นแทนด้วย 1 ปี แต่ก็เห็นว่าข้อมูลในแต่ละเดือนมีความแตกต่างกันมากโดยเฉพาะเมื่อเอามาแสดงทุกจังหวัด ทำให้แปลผลหรือนำมาสรุป trend การเปลี่ยนแปลงยาก จึงตัดสินใจว่าจะ plot เปรียบเทียบข้อมูลน้ำฝนเฉลี่ยเป็นรายปีของแต่ละจังหวัด ตั้งแต่ปีคศ. 1987 – 2022 แทน แต่ตอนนี้ใน dataframe ที่ได้มาจาก groupby YrProvince จะประกอบด้วยข้อมูลที่แสดง Year, Province และค่าเฉลี่ยน้ำฝน ไม่ได้

สิ่งที่ทำ: ทำการสร้าง dict ขึ้นมาเพื่อใส่ ข้อมูล Year, Province และค่าเฉลี่ยน้ำฝน โดยข้อมูล Year และ Province ทำการเช็คให้มั่นใจว่าไม่มีการเอาชื่อซ้ำมาโดยใช้ operation unique() ส่วนค่าเฉลี่ยน้ำฝนก็ใช้ operation mean() เพื่อให้ได้ค่าเฉลี่ยมา แล้วทำการสร้าง dataframe จาก dict เหล่านั้น

7. เมื่อทำการ visualize data เป็นค่าเฉลี่ยน้ำฝนรายปีของแต่ละจังหวัดแล้วพบว่ามีบางจังหวัดที่มีค่าที่สูงกว่าค่าจากจังหวัดอื่น ๆ ทำให้อยากให้มีการเปรียบเทียบค่านี้รวมของทุกจังหวัดกับค่าเฉลี่ยข้อมูลน้ำฝนของทั้งประเทศ
   สิ่งที่ทำ: ทำการสร้าง column ใหม่ขึ้นมาชื่อ binYear เพื่อใช้เป็นค่าที่เป็นตัวแทนช่วงปีที่เราหาค่าเฉลี่ยที่เป็น moving average และทำการสร้าง column ชื่อ Year_int ขึ้นมาเพื่อแปลงข้อมูล Year ให้เป็น type int และทำการสร้าง column ชื่อ DeMeanRain ขึ้นมาเพื่อมารองรับค่าส่วนต่างระหว่างค่าเฉลี่ยน้ำฝนของจังหวัดนึงในปีนั้น (‘Average Province’) กับค่าเฉลี่ยน้ำฝนรายปีของทั้งประเทศ ([‘Average Province’].mean())
8. ทำการ sort ข้อมูลดูแล้วพบว่าจังหวัดที่มีปริมาณน้ำฝนเฉลี่ยรายปีสูงกว่าค่าเฉลี่ยของทั้งประเทศได้แก่ ตราด ระนอง พังงา จันทบุรี นราธิวาส ภูเก็ต และ นครศรีธรรมราช เป็นต้น ซึ่งล้วนแต่เป็นจังหวัดที่ติดกับชายฝั่งทะเล ทำให้ตัดความคิดที่จะต้องการนำเสนอข้อมูลเฉพาะจังหวัดที่มีปริมาณน้ำฝนเฉลี่ยสูงสุดในประเทศไทยออกไป
   เนื่องจากอาจจะมีความ bias -ของข้อมูลอันเนื่องมาจากตำแหน่งที่ตั้งทางภูมิศาสตร์ เนื่องจากปริมาณน้ำฝนจะแปรผันตรงกับปริมาณการระเหยของน้ำในพื้นที่ใกล้เคียงด้วย ทำให้จังหวัดที่อยู่ติดกับทะเลมีโอกาสที่จะมีปริมาณฝนตกมากกว่าจังหวัดที่ไม่อยู่ติดกับน้ำทะเล ทำให้คิดว่าจำเป็นจะต้องแสดงผลข้อมูลทั้งหมด ทั้งชุดข้อมูลรวมจังหวัดที่ไม่อยู่ติดน้ำทะเลด้วย
   ปัญหาที่พบ: เวลา visaualization ชื่อที่ถูกนำมากำกับใน data label เป็นภาษาไทย แสดงผลออกมาเป็นกล่อง
   สิ่งที่ทำ: พยายามหาวิธีที่จะโหลด Thai font เข้ามา แต่ไม่ประสบความสำเร็จ
9. ทำการ visualization แล้วพบว่ากราฟที่ได้จากบางจังหวัดมีลักษณะการตัดของข้อมูล (ข้อมูลบางปีหายไป) หรือเป็นเส้นที่ไม่ได้เริ่มจากปีเริ่มต้นคือคศ.1987
   สิ่งที่ทำ: ทำการเช็คดูว่ามีข้อมูลจังหวัดไหนบ้างที่ข้อมูลไม่ครบ พบว่าจังหวัดเช่น บึงกาฬ ยโสธร และอำนาจเจริญมีข้อมูลแค่ 3 ปี จังหวัดนครนายก สมุทรสงคราม และอุทัยธานีมีข้อมูลแค่ 7 ปี หนองบัวลำภูมีข้อมูลแค่ 8 ปี เป็นต้น จึงตัดสินใจใส่เงื่อนไขในการ visualization ว่าจะแสดงเฉพาะข้อมูลจากจังหวัดที่มีข้อมูลครบ 36 ปี (ตั้งแต่คศ. 1987-2022)
10. ข้อมูล visualization ของข้อมูลค่าเฉลี่ยความแตกต่างของประมาณน้ำฝนรายปีของจังหวัดที่มีค่าสูงไม่โดดเด่น
    สิ่งที่ทำ: ทำการกำหนดสีของเส้นที่มาจากจังหวัดที่มีค่าเฉลี่ยน้ำฝนรายปีสูงกว่าค่าเฉลี่ยความแตกต่างของประมาณน้ำฝนของทั้งประเทศของทั้งประเทศ (ส่วนต่างมีค่ามากกว่า 120) ให้มีสีแดงและมีความหนาของเส้นมากกว่าข้อมูลจากสถานีอื่น ๆ (กำหนดให้เส้นเป็นสีเทา และบางกว่า) นอกจากนี้ยังได้กำหนดให้ลำดับของเส้นสีเทา (zorder =1) อยู่บนเส้นสีแดง (zorder =0) เพื่อให้เห็นความแตกต่างชัดยิ่งขึ้น
11. ข้อมูลค่าเฉลี่ยความแตกต่างของประมาณน้ำฝนของทั้งประเทศ ที่เป็น moving average เทียบกับคาบทุก 10 ปี (binYear =10) เป็นเส้นที่หยาบเกินไป อาจจะทำให้มองเปรียบเทียบกับค่าเฉลี่ยของจังหวัดยาก
    สิ่งทีทำ: เปลี่ยนค่าคาบของ binYear เป็น 5 ปี

12. ข้อมูลค่าเฉลี่ยความแตกต่างของประมาณน้ำฝนของทั้งประเทศ ที่เป็น moving average เทียบกับคาบทุก 5 ปี (binYear =5) และกำหนดให้ใช้ค่าปีที่มาแทนในการ plot เป็นค่า max (เช่นคาบ 1985 – 1990 จะใช้ค่า 1990 มา plot) นั้นมีความแหว่งในช่วงแรกของข้อมูล เส้นไม่ครอบคลุมถึงข้อมูลในช่วงแรกของ data ทั้งหมด

    สิ่งที่ทำ: เปลี่ยนไปกำหนดให้ใช้ค่าปีที่มาแทนในการ plot เป็นค่า min (เช่นคาบ 1985 – 1990 จะใช้ค่า 1985 มา plot) และพบว่าเส้นที่ plot ครอบคลุมข้อมูลทั้งหมดได้ดีกว่าการใช้ค่า max

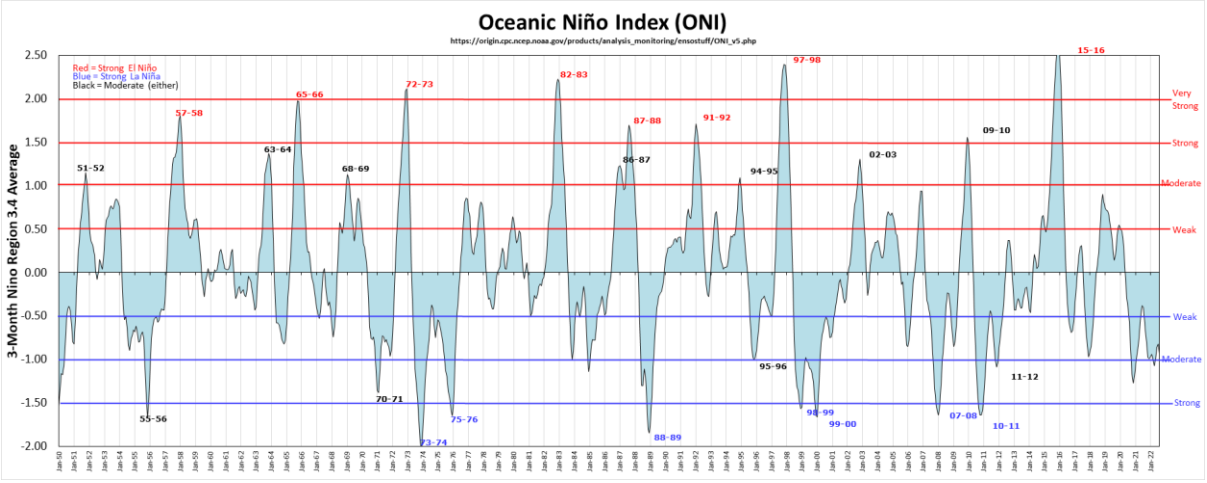
13. ปีที่แสดงในแกนนอน (xtickaxis) แสดงออกมาไม่ถูก ควรที่จะแสดงปีเป็น int แต่แสดงเฉพาะปี 1990-1994

    สิ่งที่ทำ: กำหนดปีที่ที่แสดงในแกนนอนเป็น labelyear จาก 1987 ถึง 2022 โดยเพิ่มทีละปี (1987, 2023, 1)

14. เนื่องจากคำถามวิจัยมีดังนี้

    (1) Can we correlate rain fall data in Thailand to el nino or la nina index?
    (2) What is the la nina index of the year with high anomaly (high rainfall) in Thailand?
    (3) What is the la nina index of 2022 based on rainfall data (8 months data)?
    (4) Is there a long-term trend of increasing rainfall? Could it be related to global warming?

    ซึ่งข้อ (1) – (3) จำเป็นต้องอ้างอิงกับค่า el nino index ซึ่งที่นิยมใช้อ้างอิงกันคือการอ้างอิงกับ Nino 3.4 region (ดูข้อมูลเพิ่มเติมได้จาก https://ggweather.com/enso/oni.htm) ซึ่งข้อมูลที่หาได้จะเป็นรูปกราฟดังแสดงด้านล่าง ไม่มีข้อมูลเป็น csv หรือตารางค่าที่สามารถนำมาพล็อตเปรียบเทียบได้



    สิ่งที่ทำ: ทำการอ่านค่าโดยประมาณ เป็นค่า ONI ตัวแทนของแต่ละปี และทำเป็นตาราง csv เพื่อนำมาพล็อตเปรียบเทียบกับข้อมูลน้ำฝนเฉลี่ยในประเทศไทย แต่ข้อมูลกราฟที่ได้มาก็จะไม่ capture ค่า ONI ที่แท้จริง และอาจจะ mis-represent ในบางจุด

15. ค่า ONI ที่นำมาพล็อต เวลาเอามาพล็อตร่วมกันในกราฟเดียวกันกับปริมาณน้ำฝนเฉลี่ยของไทยซึ่งมีการใช้แกน y ในการ plot ไปกับ 2 data set แล้ว การมาเพิ่มค่าที่ 3 ในแกน y จึงทำให้การกำหนดแกนทำได้ยาก พอจะทำให้เป็นกราฟย่อยในกราฟปริมาณน้ำฝนเฉลี่ย ก็ต้องใช้เวลาเพิ่มมากขึ้น ด้วยความที่ยังไม่คล่องกับการพล็อต matplotlib ทำให้ทำไม่ทัน

    สิ่งที่ทำ: ทำการแยกพล็อตค่า ONI ใน plot ใหม่ (ในอนาคตควรพยายามพล็อตในกราฟเดียวกัน)

16. ค่า ONI ที่นำมาพล็อตนั้นค่าที่เป็น + แสดงถึงปีที่เป็น el nino (แล้ง) และปีที่เป็น - แสดงถึงปีที่เป็น la nina (ฝนตกหนัก และฤดูหนาว หนาวมาก) เวลาเอามาเทียบกับกราฟที่พล็อตแสดงปริมาณน้ำฝนเฉลี่ย (ค่าสูง = ปีที่เป็น la nina) ทำให้แปลผลหรือเห็นความสัมพันธ์ได้ยาก

    สิ่งที่ทำ: ทำการแก้ค่า ONI ในตาราง csv ให้เป็นค่า negative ONI เพื่อที่เวลานำมาพล็อตเปรียบเทียบกับข้อมูลน้ำฝนเฉลี่ยในประเทศไทย ค่าที่สูงคือค่าที่แสดงถึงปีที่เป็น la nina เหมือนกัน