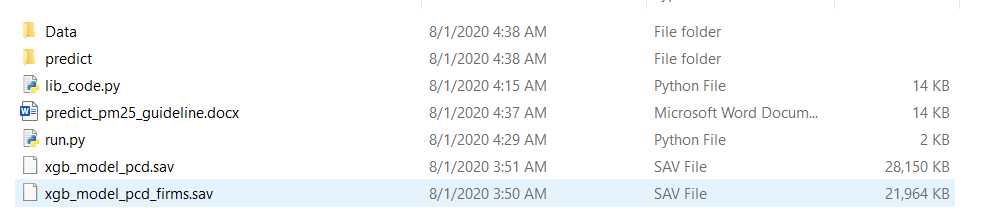
1. เริ่มต้นใน serve ต้อง install package

* Numpy
* Pandas
* Selenium
* Webdriver\_manager
* beautifulsoup4
* Matplotlib
* Pickle
* Pip install numpy pandas selenium webdriver\_manager beautifulsoup4 matplotlib pickle

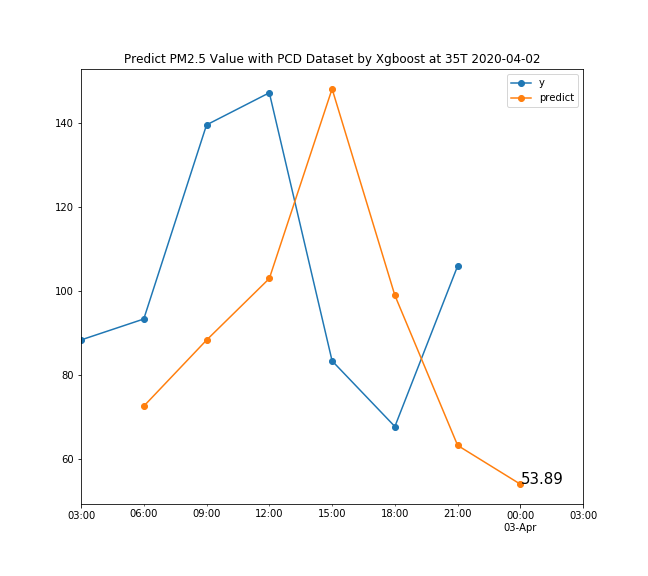
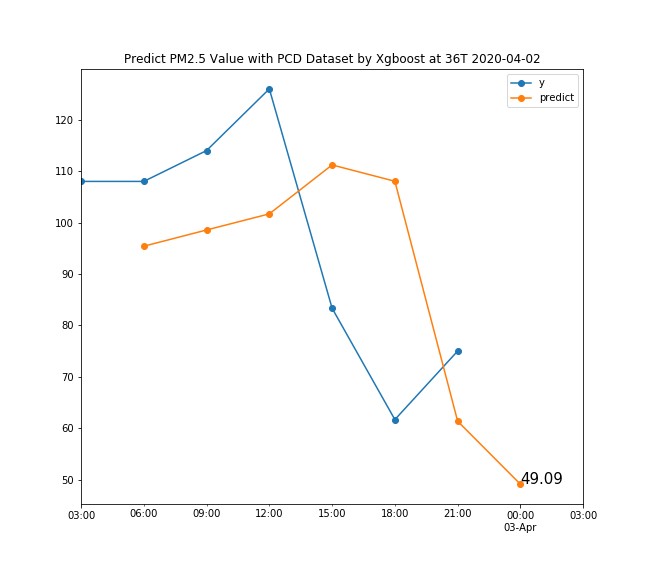


2. ตามรูปด้านบน จะมีทั้งหมด 2 file และ 2 folder

Folder data -> เราใช้สำหรับเก็บข้อมูลที่ทำการ scrapping มาจากหน้าเว็บของทางกรมควบคุมมลพิษ ถ้าวิมสามารถ query ข้อมูลส่วนนั้นมาได้โดยตรงก็ไม่ต้องสนใจเพราะจะไม่ได้ใช้

Folder predict -> เป็น folder ที่เราใช้เก็บ graph ที่ได้จากการทำนายนะ

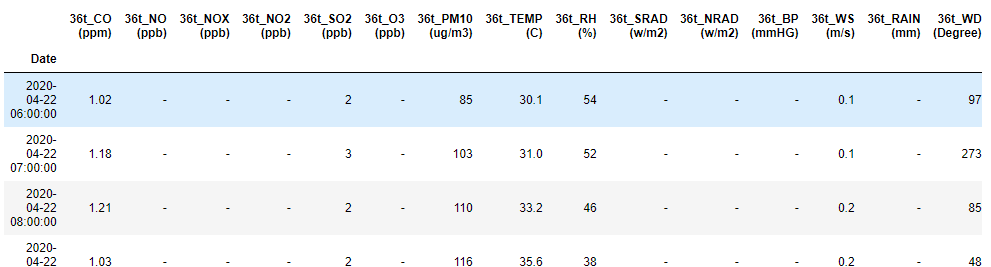
\*\* ถ้าจะทำนายโดยใช้โมเดลเรา สถานีอื่นๆ ที่ไม่ใช่ 35t กับ 36t ผลลัพท์อาจจะไม่สอดคล้องกันเท่าไหร่นะ เพราะตัวโมเดลที่เราเทรนเราใช้แค่ 35t, 36t สองสถานีเท่านั้น ขนาดลองทำนายของสองสถานีที่ใช้ทำนายผลลัพท์ยังไม่ค่อยแม่นเลย ตามตัวอย่างด้านล่างโดยค่าที่ทำนายได้ค่อนข้างจะใกล้เคียงกับของชั่วโมงก่อนหน้า ตามรูปด้านล่าง\*\*\* (บอกไว้ก่อนน)



การเขียนโค้ดตัวไฟล์ python แบ่งเป็น 2 file

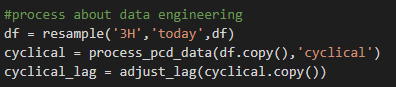
Lib\_code.py -> ไฟล์นี้เป็นไฟล์เก็บฟังชันก์ต่างๆ โดยจะถูกเรียกใช้ใน run.py ถ้าอยากแก้ไขอะไรเข้าไปแก้ได้

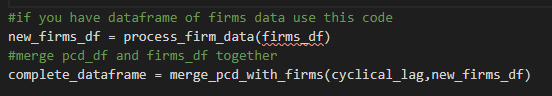
Run.py -> การทำงานของ run.py คือส่วนใหญ่ใช้พวก dataframe (dataframe ก็คือ รูปแบบของการจัดเก็บข้อมูลคล้ายๆ excel แบบนั้นอะ เป็นตาราง มี row มี columns )ในการเก็บข้อมูลอะ ตัวอย่างก็รูปด้านล่าง



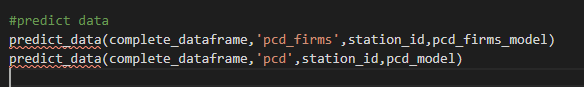
1. ตอนแรกเราจะให้มันไปเก็บข้อมูลหน้าเว็บของกรมควบคุมมลพิษ ส่วนนี้เรียกว่า web scrapping ซึ่งถ้าทำเสร็จแล้วจะได้ dataframe ออกมาหนึ่งอัน ตรง 

โดยที่ถ้าวิมสามารถ query ข้อมูลมาได้โดยตรงก็ไม่จำเป็นต้องทำ web scrapping ข้ามขั้นนี้ไปเลยแต่ว่าต้องเอาข้อมูลที่ query มาใส่ลงไปใน dataframe นะ (ใช้ pandas)

1. ต่อไปขั้นตอนของการ feature engineering โดยจะเอา dataframe ที่ได้อะไปปรับค่าข้อมูลในแต่ละ feature ให้ตัวโมเดลของเราอะ สามารถอ่านได้ และก็เติมข้อมูลในส่วนที่หายไปในแต่ละ feature ด้วยและอีกอย่างคือปรับข้อมูลเหล่านั้นจากรายชั่วโมงเป็นรายสามชั่วโมง (ลืมบอก โมเดลเราทำนายค่าฝุ่นในรายสามชั่วโมงนะอย่างเช่นเวลาเที่ยงคืนทำนายค่าตอนตีสามประมาณนี้)
2. ขั้นที่สามเป็นขั้นตอนการปรับข้อมูลของ firms ส่วนนี้อะเหมือนกับข้อ 1 คือให้วิมดึงข้อมูลของ firms ในระยะไม่เกินซัก 500 กิโมเมตรก็ได้ หรือไม่งั้นก็ดึงข้อมูลล่าสุดย้อนหลังมา 7 วันก็ได้ ตัวโปรแกรมมันจะคำนวณระยะทางที่ห่างจากเชียงใหม่แล้วเอาจุดที่ใกล้ๆ มาพิจารณาแล้วมันจะปรับให้เป็นสามชั่วโมงด้วย คือดึงมาแล้วต้องทำให้เป็น dataframe ด้วยนะเหมือนข้อ1
3. เสร็จแล้วก็จะรวมข้อมูลทั้งสองเข้าด้วยกันทั้ง pcd จากกรมควบคุมมลพิษ และก็ firms รูปด้านล่างาคือข้อ 4 และ5 ที่มันแดงเพราะว่า firms\_df เราไม่ได้ใส่ วิมต้อง query ข้อมูลมาใส่ dataframeแล้วเอาเข้า functionนี้



1. จะเป็นการ load model เรา train model ให้แล้วใช้ pickle โหลด model เข้ามาโดยจะมี 2 model ด้วยกัน
2. xgb\_model\_pcd\_firms.sav เป็น model จาก algo ชื่อ extreme gradient boosting ซึ่งเทรนมาจากข้อมูล pcd กับ firms รวมกันของจุด 35t, 36t ในปี 2016- 2018
3. xgb\_model\_pcd.sav เป็น model จาก algo ตัวเดียวกัน แต่ว่าเทรนมาจากข้อมูลของ pcd อย่างเดียว ของจุด 35t, 36t ในปี 2016- 2018



คือที่ทำนายแยกสอง model เนื่องจากว่าบางทีข้อมูลจาก firms มันช้า อย่างเช่น ข้อมูลจาก pcd มันมีถึงตอน 7 โมงเช้า แต่ว่า firms ยังมีถึงตี 2 เมื่อนำมารวมกันทำให้ pcd มันมีข้อมูลถึงตี 2 ดังนั้นจึงทำนาย 3 ชั่วโมงข้างหน้าไม่ได้ (อันนี้เป็นปัญหาที่เราเจอนะ แต่ถ้าวิมดึง realtime ได้ก็ไม่มีปัญหาอะไร) ก็เลยทำแยก

ฟังก์ชัน Predict\_data มันจะเอาข้อมูลไปทำนายแล้วนำไป plot graph เก็บใน predict นะ