การประเมินคุณภาพอากาศจากภาพถ่ายดาวเทียมโดยใช้การเรียนรู้ของเครื่อง

AIR QUALITY ASSESSMENT BASED ON SATELLITE IMAGES USING MACHINE LEARNING

ณัฐเดช วิจารณกุล

NATTADET VIJARANAKUL

วิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2565

KMITL-2022-SC-M-002-127

AIR QUALITY ASSESSMENT BASED ON SATELLITE IMAGES USING MACHINE LEARNING

NATTADET VIJARANAKUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2022
KMITL-2022-SC-M-002-127

หัวข้อวิทยานิพนธ์ การประเมินคุณภาพอากาศจากภาพถ่ายดาวเทียมโดยใช้การเรียนรู้ของเครื่อง

ชื่อนักศึกษา ณัฐเดช วิจารณกุล

รหัสนักศึกษา 61605052

ปริญญา วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

ภาควิชา วิทยาการคอมพิวเตอร์

พ.ศ. 2565

อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร. กุลสวัสดิ์ จิตขจรวานิช

บทคัดย่อ

ปัจจุบันช่วงฤดูหนาวของประเทศไทย (พฤศจิกายน - กุมภาพันธ์) กรุงเทพและปริมณฑลประสบปัญหามลพิษทางอากาศ เช่น PM 2.5 โดยทั่วไปแล้วการวัดค่าคุณภาพอากาศจะทำโดยการตั้งสถานีตรวจวัดไว้ยังจุดต่าง ๆ แต่การตั้งสถานีตรวจวัดมีข้อเสียคือการวัดไม่สารถกำหนดได้อย่างแน่ชัดว่าสามารถใช้แทนข้อมูลได้ในระยะรัศมีกี่เมตร ดังนั้นทางผู้จัดทำจึงได้เสนอวิธีการใช้ข้อมูลที่ครอบคลุมไปยังพื้นที่ที่เป็นวงกว้าง เช่นภาพถ่ายดาวเทียม โดยภาพถ่ายดาวเทียมที่นำมาใช้ในงานวิจัยนี้คือภาพถ่ายของดาวเทียม Landsat 8 และวิธีการตรวจวัดค่าคุณภาพอากาศจากภาพถ่ายดาวเทียมนั้นเราได้ใช้การเรียนรู้ของเครื่องเข้ามาเป็นตัวตรวจวัด โดยการเรียนรู้ที่นำมาใช้ในงานวิจัยนี้เกิดจากการคัดเรื่องตัวแบบที่เหมาะสมระหว่าง การวิเคราะห์ความถดถอยเชิงเส้น, ต้นไม้ตัดสินใจ, กฎของเบย์อย่างง่าย, K Nearest Neighbors, Random Forest และ Gradient Boosting ซึ่งผลลัพธ์ของงานวิจัยนี้คือได้ตัวแบบการเรียนรู้ของเครื่องแบบผสม (Hybrid Model) ที่เกิดจากการผสานการทำงานกันของตัวแบบคัดแยก (Classification Model) และตัวแบบถดถอย (Regression Model) ซึ่งตัวแบบการเรียนรู้ของเครื่องแบบผสมนั้นให้ผลลัพธ์ที่ Mean Absolute Error = 8.3864 และ R^2 = 0.7499 ซึ่งเป็นผลลัพธ์ที่ดีกว่าการใช้ตัวแบบที่เกิดจากการใช้เพียงตัวแบบถดถอยเพียงอย่างเดียว (Pure Regression Model) โดยตัวแบบถดถอยที่ให้ผลลัพธ์ที่ดีที่สุดคือ K Nearest Neighbors เมื่อค่า K = 1 ที่ผลลัพธ์ Mean Absolute Error = 8.9559 และ R^2 = 0.7183

Thesis Title	Air Quality Assessment Based On Satellite Images Using Machine Learning
Student Name	Nattadet Vijaranakul
Student ID	61605052
Degree	Master of Science (Computer Science)
Department	Computer Science
Year	2022
Thesis Advisor	Asst.Prof.Dr.Kulsawasd Jitkajornwanich

Abstract

During the winter in Thailand (November – February), Bangkok and its metroplex face air pollution problem, i.e., PM 2.5. The air quality assessment method used is to implement physical air quality measure devices at specific locations. This method, however, has a limitation about the coverage areas especially remote locations where the air quality were not assessed properly. To solve this problem, we utilize additional data, i.e., satellite images, that can cover more and wider areas. In this work we propose a methodology that incorporates satellite images for air quality assessment with machine learning techniques. Satellite images used in this work are collected from Landsat 8. And for machine learning models, we use Linear Regression, Decision Trees, Naïve Bayes, K-Nearest Neighbors, Random Forest, and Gradient Boosting. In our research, we also experiment with combined classification techniques and regression techniques that we call “Hybrid Model”. The Hybrid Model has performance with Mean Absolute Error = 8.3864 and R2 = 0.7499. Its performance is better than using only regression model that we call “Pure Regression Model”. The best Pure Regression Model is K-Nearest Neighbors when K = 1 which has performance of Mean Absolute Error = 8.9559 and R2 = 0.7183.

กิตติกรรมประกาศ

การที่วิทยานิพนธ์ฉบับนี้สามารถเกิดขึ้นมาได้เนื่องจากความอนุเคราะห์ของบุคคลหลายฝ่าย ซึ่งแต่ละท่านก็ให้ความอนุเคราะห์ต่าง ๆ ซึ่งกระผมจะขอกราบขอบพระคุณทุกท่านมาในโอกาสนี้ ดังนี้

กราบขอบพระคุณ สำนักงานธรณีวิทยาสหรัฐอเมริกา (USGS) รวมทั้งเจ้าหน้าที่ประสานงานของหน่วยงาน ที่อนุเคราะห์ข้อมูลภาพถ่ายดาวเทียม Landsat 8 ซึ่งเป็นหัวใจของงานวิจัยในวิทยานิพนธ์นี้

กราบขอบพระคุณ กรมควบคุมมลพิษ กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม ที่อนุเคราะห์ข้อมูลคุณภาพอากาศ รวมถึงเจ้าหน้าที่ที่คอยแนะนำการวิธีการใช้ข้อมูลคุณภาพอากาศ ซึ่งเป็นอีกหนึ่งหัวใจหลักในวิทยานิพนธ์นี้

กราบขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.สายชล ใจเย็น อาจารย์ประจำคณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ซึ่งเป็นอดีตอาจารย์ประจำภาควิชาวิทยาการคอมพิวเตอร์ และคณะกรรมการสอบหัวข้อวิทยานิพนธ์นี้ ซึ่งได้แนะนำวิธีการใช้งาน, การปรับแต่ง รวมถึงการวิเคราะห์ข้อมูลต่าง ๆ ที่ได้รับจากการเรียนรู้ของเครื่อง

กราบขอบพระคุณท่าน ดร.ภาณุ เศรษฐเสถียร และ ดร.สยาม ลววิโรจน์วงศ์ จากสำนักงานพัฒนาเทคโนโลยีอวกาศและภูมิสารสนเทศ (องค์การมหาชน) ที่แนะนำการใช้งานภาพถ่ายดาวเทียม และการประกอบการทำงานของการเรียนรู้ของเครื่องกับภาพถ่ายดาวเทียม รวมถึงวิธีการแปรผลลัพธ์ที่ได้จากการเรียนรู้ของเครื่องด้วย

กราบขอบพระคุณท่านผู้ช่วยศาสตราจารย์ ดร.กุลสวัสดิ์ จิตขจรวานิช อาจารย์ที่ปรึกษา ที่คอยให้คำแนะนำในการตีพิมพ์งานวิจัย และเป็นส่วนช่วยให้เกิดวิทยานิพนธ์ฉบับนี้

และสุดท้ายนี้ขอกราบพระคุณคุณพ่อผู้ล่วงลับ และคุณแม่ที่คอยให้กำลังใจสนับสนุนการทำงานเพื่อให้วิทยานิพนธ์นี้สำเร็จสมบูรณ์ออกมาได้

นายณัฐเดช วิจารณกุล

ผู้จัดทำวิทยานิพนธ์

สารบัญ

สารบัญ

สารบัญตาราง

สารบัญรูป

คำย่อ/สัญลักษณ์

คำย่อ, สัญลักษณ์	ความหาย
TP	ค่า True Positive
TN	ค่า True Negative
FP	ค่า False Positive
FN	ค่า False Negative
X_y	ค่า x (หมายถึงค่า TP TN FP หรือ FN) ของ class y
MAE	ค่า Mean Absolute Error
R^2	ค่า Coefficient of Determination
Level 2 (SR)	Level 2 Surface Reflect

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในช่วงกลางฤดูหนาวของประเทศไทยของทุกปี (ธันวาคม - มกราคม) กรุงเทพมหานครได้ประสบปัญหาวิกฤตมลพิษทางอากาศ ซึ่งเป็นปัญหาสำคัญและส่งผลกระทบต่อการใช้ชีวิตประจำวันของประชาชน โดยวิธีการตรวจวัดคุณภาพอากาศนั้น ในปัจจุบันได้มีการนำเครื่องตรวจวัดไปติดตั้งตามจุดต่างๆ ซึ่งยังมีข้อเสียคือเราไม่สามารถรู้ได้ว่าค่าที่อ่านได้จากเครื่องตรวจวัดในหนึ่งจุดนั้นสามารถแทนค่าค่าตรวจวัดในรัศมีโดยรอบได้ในรัศมีเท่าไรแน่นอน เพราะค่าตรวจวัดที่ได้จะมีความแตกต่างกันไปตามสภาพแวดล้อมในแต่ละพื้นที่ ด้วยเหตุนี้ทางผู้วิจัยจึงเห็นว่าหากนำภาพถ่ายดาวเทียมมาใช้จะครอบคลุมพื้นที่ได้มากกว่า(ทั้งภาพ) และนำมาใช้ในการตรวจวัดคุณภาพอากาศแทนด้วยโครงสร้างข้อมูลของภาพถ่ายดาวเทียมมีความหลากหลายของช่วงคลื่นในหนึ่งภาพ เพื่อให้การวิเคราะห์คุณภาพอากาศทำได้อย่างมีประสิทธิภาพ เราได้ใช้เครื่องมือที่จะนำมาใช้ในการถอดข้อมูลจากภาพ Open Data Cube (ODC) ซึ่งเป็น Framework ในการเก็บข้อมูล, เรียกใช้ข้อมูล(Query), และแสดงผลภาพถ่ายดาวเทียมจากฐานข้อมูล สำหรับเหตุผลที่เลือก Open Data Cube (ODC) ในการวิจัยครั้งนี้เนื่องจากมีการติดตั้ง และใช้งานได้ง่าย รวมถึง Open Data Cube (ODC) ได้ถูกพัฒนาโดยใช้ภาษา Python ซึ่งเป็นภาษาที่ใช้ในการวิเคราะห์ข้อมูลในศาสตร์ของวิทยาศาสตร์ข้อมูลอย่างแพร่หลาย จึงทำให้ง่ายต่อการนำข้อมูลไปวิเคราะห์และประยุกต์วิธี เพื่อศึกษา, เรียนรู้เครื่องมือ ODC Framework หรือไปสร้างตัวแบบสำหรับงานด้านวิทยาศาสตร์ข้อมูล

1.2 วัตถุประสงค์ของงานวิจัย

1) เสนอวิธีใหม่ในการตรวจวัดคุณภาพอากาศ ที่นอกเหนือจากการตั้งสถานีตรวจวัดไปยังจุดต่างๆ โดยการใช้ข้อมูลภาพถ่ายดาวเทียม

2) ศึกษาการใช้งานข้อมูลภาพถ่ายดาวเทียมสำรวจโลก

3) ศึกษาการประยุกต์ใช้งานภาพถ่ายดาวเทียม กับการเรียนรู้ของเครื่อง

4) ศึกษาการประยุกต์ใช้วิธีการทางการเรียนรู้ของเครื่องในแบบต่าง ๆ

1.3 ขอบเขตของงานวิจัย

1) พื้นที่การศึกษาโดยภาพถ่ายดาวเทียม จะทำในบริเวณกรุงเทพฯ และปริมณฑล ที่มีเครื่องมือตรวจวัดที่สามารถเข้าถึงได้ และใช้เป็นแหล่งอ้างอิง

2) ภาพถ่ายดาวเทียมที่ใช้จะใช้เฉพาะภาพ Level 2 ของภาพถ่ายดาวเทียม Landsat 8 รองรับการประเมินคุณภาพอากาศบนพื้นที่ไม่ถูกเมฆบังบนภาพถ่ายดาวเทียมเท่านั้น

3) ฐานข้อมูลที่นำมาจัดเก็บภาพถ่ายดาวเทียมจะใช้เพียงระบบฐานข้อมูล PostgreSQL และทำงานผ่าน API Open Data Cube เท่านั้น ทั้งการจัดเก็บ และการเรียกใช้งาน

4) งานวิจัยนี้จะไม่มองถึงความสามารถในการประมวลผลของเครื่อง เช่น เวลาที่ใช้ประมวลผล หรือวิธีการทางการโปรแกรม

1.4 สมมติฐานการทดลอง

1) ระบบสามารถทำงานได้ทุกระบบที่รองรับการทำงานของโปรแกรมที่ถูกพัฒนาโดยภาษา Python

2) ตัวแบบสำหรับทำนาย การประเมินคุณภาพอากาศที่มีค่า Mean Absolute Error น้อยกว่า 25 (ไม่เกินช่วงที่สั้นที่สุดของระดับคุณภาพอากาศ)

1.5 ข้อจำกัด

1) ภาพถ่ายดาวเทียมที่จะใช้งานได้ดี คือ ภาพถ่ายที่ไม่มีสิ่งรบกวน เช่น การบดบังของเมฆ หรือภาพที่เกิดการกระเจิงของแสง

2) ระบบจะทำงานกับเครื่องที่รองรับการทำงานของโปรแกรมที่ถูกพัฒนาขึ้นด้วยภาษา Python เท่านั้น

3) ระบบจะทำงานได้กับเครื่องที่สามารถใช้งานระบบฐานข้อมูล PostgreSQL ได้ เท่านั้น

3) ระบบจะทำงานได้บนเครื่องที่สามารถรองรับการทำงานด้วยไลบรารี่ Open Data Cube, Pandas, Scikit-Learn และ ไลบราลีอื่น ๆ ที่เกี่ยวข้อง เท่านั้น

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้ ตัวแบบสำหรับทำนาย สำหรับการทำนายค่าคุณภาพของอากาศที่มี Mean Absolute Error ต่ำกว่า 25

2) ได้วิธีวัดค่าคุณภาพอากาศที่สามารถครอบคลุมไปยังพื้นที่ใดพื้นที่หนึ่ง

3) ได้วิธีใช้ข้อมูลของดาวเทียมสำรวจพื้นผิวโลกแบบใหม่

1.7 ขั้นตอนการดำเนินงาน

1) ศึกษาความเป็นไปได้ของงานวิจัย และงานวิจัยอื่นๆ ที่ใกล้เคียงจาก Scopus/ISI

2) เก็บและศึกษาข้อมูลที่จะนำมาใช้งาน

3) ศึกษา Software ที่จะนำมาใช้งาน ทั้งฐานข้อมูล และ API

4) ศึกษา ตัวแบบสำหรับทำนาย และประเมินข้อดี/ข้อเสียของแต่ละ ตัวแบบสำหรับทำนายที่จะนำมาทดลอง

5) ทำการทดลองและปรับโครงสร้าง ตัวแบบสำหรับทำนาย ตามความเหมาะสม

6) บันทึกผลการทดลองและเปรียบเทียบประสิทธิภาพ

7) นำ ตัวแบบสำหรับทำนาย ที่มีประสิทธิภาพสูงสุดไปพัฒนาใช้จริง

1.8 อุปกรณ์ที่ใช้

1.8.1 Hardwares

1) Personal Computer (PC) ราคาประมาณ 30,000 บาท

- CPU AMD Ryzen5 3600
- VGA NVIDIA RTX 2060 1920 CUDA Cores Memory GDDR6 6 GB
- RAM 32 GB Bus 2666
- SSD M.2 PCIe NVMe 500 GB (สำหรับใส่ OS, และ Library ต่าง ๆ)
- HDD 2 TB 7200 RPM (ใช้สำหรับใส่ไฟล์โปรแกรม และภาพถ่ายดาวเทียม)

1.8.2 Softwares

1) ระบบปฏิบัติการ Windows 10
2) PostgreSQL (RDBMS สำหรับเก็บข้อมูล)
3) Python 3
4) Open Data Cube และสภาพแวดล้อมต่าง ๆ ที่ ODC ต้องการ
5) Scikit-Learn
6) Pandas
7) Google Colab (ใช้สำหรับ Train ข้อมูล)
8) Microsoft Excel เพื่อทำการตรวจสอบและคัดกรองข้อมูลในเบื้องต้น

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

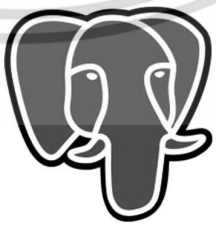
2.1 ฐานข้อมูลสำหรับข้อมูลเชิงพื้นที่

ฐานข้อมูลและตัวจัดการเกี่ยวกับข้อมูลเชิงพื้นที่นั้นมีหลายตัว เช่น PostGIS, QGIS ที่ใช้ร่วมกับฐานข้อมูล PostgreSQL หรือ GoogleAWS ที่ใช้ฐานข้อมูลของ Amazon เป็นต้น ซึ่งในงานวิจัยนี้จะใช้ฐานข้อมูล PostgreSQL ร่วมกับตัวจัดการข้อมูลเชิงพื้นที่ Open Data Cube

2.1.1 ระบบจัดการฐานข้อมูล PostgreSQL

โพสเกรสคิวแอล (หรือนิยมเรียกกันว่า โพสเกรส) เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database) แบบ Free-Open Source ถูกพัฒนาขึ้นโดย PostgreSQL Global Development Group โดยมีชื่อเดิมว่า Postgres ซึ่งได้มาจากโครงการที่เป็นรากฐานของการพัฒนาฐานข้อมูลนี้จากมหาวิทยาลัยแคลิฟอร์เนีย เบิร์คลีย์ ในชื่อว่า Post Ingres และภายหลังได้เปลี่ยนชื่อเป็น PostgreSQL เพื่อให้สื่อถึงภาษา SQL ซึ่งเป็นภาษาที่ใช้ร่วมกับฐานข้อมูลเชิงสัมพันธ์ (Relational Database) โดย PostgreSQL มี Transaction Features ต่างๆ ตามหลักทั่วไปของฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เช่น ACID Properties, คีย์นอก (Foreign Key) เป็นต้น เครื่องหมายการค้าของ PostgreSQL มีลักษณะเป็นรูปหัวช้างสีฟ้า

PostgreSQL เป็นฐานข้อมูลที่นิยมใช้ในงานเกี่ยวกับข้อมูลภูมิสารสนเทศ (GIS Data) โดยมีการติดตั้งร่วมกับ PostGIS เป็นต้น PostgreSQL เป็นฐานข้อมูลที่มีการติดตั้งที่ง่าย เหมาะสำหรับผู้ที่ไม่สันทัดทางด้านคอมพิวเตอร์มากนัก



รูปที่ 2.1 เครื่องหมายการค้าของ PostgreSQL [1]

2.1.2 Open Data Cube (ODC)

โอเพนดาต้าคิวบ์ (หรือโอดีซี) เป็น API แบบ Opensource ที่ถูกพัฒนาขึ้นโดยไลบรารีต่างๆ เกี่ยวกับภาพถ่ายดาวเทียมบนภาษา Python และ ฐานข้อมูล PostgreSQL มีหน้าที่ในการจัดการเกี่ยวกับ การจัดเก็บ เรียกใช้ รวมถึงการประมวลผลข้อมูล และแสดงผลภาพถ่ายดาวเทียม โดย Open Data Cube ได้สร้าง Interface และโปรแกรม สำหรับจัดการการจัดเก็บ เรียกใช้ และประมวลผลไว้เพื่อลดความยุ่งยากต่อผู้พัฒนาโปรแกรมในการเรียกใช้ และอีกทางหนึ่ง Open Data Cube ยังได้มีการแจกโปรแกรมตัวอย่างไว้บน Github เพื่อเป็นตัวอย่างการใช้งาน และเป็นแนวทางในการต่อยอดพัฒนาโปรแกรมของผู้ที่ต้องการจะพัฒนาด้านการใช้ข้อมูลภาพถ่ายดาวเทียม

รูปที่ 2.2 เครื่องหมายการค้าของ Open Data Cube [2]

2.2 ภาพถ่ายดาวเทียม

ซึ่งดาวเทียมที่มีใช้อยู่ในโลกของเรามีอยู่หลายประเภท เช่น ดาวเทียมโทรคมนาคม ดาวเทียมอุตุนิยมวิทยา หรือดาวเทียมสำรวจทรัพยากรบนโลก เป็นต้น ทั้งนี้จะมีดาวเทียมที่สามารถถ่ายภาพพื้นผิวหรือเหตุการณ์ต่างๆ ที่เกิดขึ้นบนโลกได้แค่บางประเภทเท่านั้น เช่น ดาวเทียมอุตุนิยมวิทยา และดาวเทียมสำรวจทรัพยากรของโลก ซึ่งดาวเทียมที่สามารถถ่ายภาพพื้นผิวของโลกได้นั้นยังมีการแบ่งออกไปตามความละเอียดเชิงพื้นที่ (Spatial Resolution) อีกด้วย ซึ่งภาพถ่ายดาวเทียมที่จะนำมาใช้ในการวิจัยครั้งนี้คือภาพถ่ายของดาวเทียม Landsat 8 (LS8) ซึ่งเป็นดาวเทียมสำรวจทรัพยากรโลกของ องค์การบริหารการบินและอวกาศแห่งชาติ สหรัฐ (NASA) และเป็นดาวเทียมที่มีความละเอียดเชิงพื้นที่อยู่ในระดับกลาง (Middle Spatial Resolution Satellite)

2.2.1 ดาวเทียม Landsat 8 [3]

รูปที่ 2.3 ดาวเทียม Landsat 8

เป็นดาวเทียมสำรวจโลกสัญชาติอเมริกา (American earth observation satellite) ที่ถูกพัฒนาขึ้นโดยบริษัท Orbital Science ร่วมมือกับ Ball Aerospace และ NASA ถูกปล่อยขึ้นสู่วงโคจรเมื่อวันที่ 11 กุมภาพันธ์ 2013 โดยมีหน้าที่ถ่ายภาพพื้นผิวของโลก โดยดาวเทียม Landsat 8 จะมีคาบการถ่ายภาพ ณ จุดเดิมทุก 16 วัน มีความละเอียดเชิงพื้นที่ (Spatial resolution) 30 เมตร และมีเครื่องมือที่ใช้ถ่ายภาพพื้นผิว (Band Suit) ต่าง ๆ ดังนี้

Spectral Band	Wavelength	Resolution	Solar Irradiance
Band 1	0.433 – 0.453 μm	30 m	2031 W/(m^2μm)
Band 2	0.450 – 0.515 μm	30 m	1925 W/(m^2μm)
Band 3	0.505 – 0.600 μm	30 m	1826 W/(m^2μm)
Band 4	0.630 – 0.680 μm	30 m	1574 W/(m^2μm)
Band 5	0.845 – 0.885 μm	30 m	955 W/(m^2μm)
Band 6	1.560 – 1.660 μm	30 m	242 W/(m^2μm)
Band 7	2.100 – 2.300 μm	30 m	82.5 W/(m^2μm)
Band 8	0.500 – 0.680 μm	15 m	1739 W/(m^2μm)
Band 9	1.360 – 1.390 μm	30 m	361 W/(m^2μm)
Band 10	10.30 – 11.30 μm	100 m	-
Band 11	11.50 – 12.50 μm	100 m	-

ตารางที่ 2.1 ตารางแสดงชุดภาพถ่ายพื้นผิวโลกของดาวเทียม Landsat 8 [3]

โดยข้อมูลที่ใช้จะเป็นภาพถ่ายพื้นผิวโลกที่ผ่านการประมวลผลบางส่วนจากผู้พัฒนาแล้ว โดยจะเรียกภาพถ่ายที่ผ่านการประมวลผลแล้วว่าภาพถ่าย Level 2 โดยภาพถ่าย Level 2 จะมี Band ดังนี้

Band	Band name	Resolution
Band 1	Coastal Aerosol	30 m
Band 2	Blue	30 m
Band 3	Green	30 m
Band 4	Red	30 m
Band 5	Near Infrared (NIR)	30 m
Band 6	Short Wave Infrared 1 (SWIR1)	30 m
Band 7	Short Wave Infrared 2 (SWIR2)	30 m

ตารางที่ 2.2 ตารางแสดงชุดภาพถ่าย Level 2 ของดาวเทียม Landsat 8 [3]

และยังมีข้อมูล Pixel_QA ซึ่งอยู่ในรูปการณ์แปรเลขฐานสองแบบ 16 bit เป็นเลขฐาน 10 ซึ่งแต่ละ Bit จะมีความหมายเรื่องการที่มีสิ่งรบกวนใดอยู่ใน Pixel นั้น ๆ

2.2.2 ค่า Vegetation Indices [4][5]

ค่า Vegetation Indices คือค่าต่าง ๆ ที่แสดงถึงดัชนีมวลพรรณ ที่มีอยู่ในภาพถ่ายดาวเทียม Pixel นั้น สามารถคำนวณหาได้จากค่า Band Near Infrared (NIR) และ Band Red ซึ่งการนำค่า Vegetation Indices มาใช้นั้นมีที่มาจากงานของ Chitrini Mozumder et al. [x] ซึ่งมีการแสดงว่าค่า Vegetation Indices ต่าง ๆ นั้นมีความแปรผันในเชิงเส้นกับค่าคุณภาพอากาศ โดย Vegetation Indices ที่จะนำมาใช้มีดังนี้

2.2.2.1 Vegetation Index (VI)

ค่า Vegetation Index (VI) เป็นค่าที่บ่งบอกว่าใน pixel นั้นๆ ของภาพถ่ายดาวเทียมมีความเป็นไปได้เท่าไรว่าจุดนั้นเป็นพืชพรรณ หาได้จาก

$$VI = NIR - RED$$

เมื่อ NIR คือค่าของ Band Near Infrared

RED คือค่าของ Band Red (คลื่นสะท้อนสีแดง)

2.2.2.2 ค่า Normalized Difference Vegetation Index (NDVI)

ค่า Normalized Difference Vegetation Index (NDVI) เป็นค่าที่บ่งบอกถึงความแข็งแรงของพืชพรรณ หรือระยะการเติบโตของพืชพรรณใน pixel นั้นๆ ของภาพถ่ายดาวเทียม หาได้จาก

$$NDVI = \frac{NIR-Red}{NIR+Red}$$

เมื่อ NIR คือค่าของ Band Near Infrared

RED คือค่าของ Band Red (คลื่นสะท้อนสีแดง)

2.2.2.3 ค่า Transformed Vegetation Index (TVI)

ค่า Transformed Vegetation Index เป็นค่าที่ไว้ประยุกต์ใช้กับพื้นที่ที่เป็นทุ่งหญ้า หาได้จาก

$$TVI = \sqrt{NDVI + 0.5}$$

2.3 ข้อมูลคุณภาพอากาศ (Air Quality Index Data) [6]

ข้อมูลคุณภาพอากาศที่จะนำมาทดลองในที่นี้เป็นข้อมูลของกรมควบคุมมลพิษ กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม ประเทศไทย โดยปกติข้อมูลคุณภาพอากาศที่ได้จะเป็นค่าเฉลี่ยคุณภาพอากาศ 24 ชั่วโมง และข้อมูลจะมีการอัพเดตทุก 1 ชั่วโมง ซึ่งข้อมูลปัจจุบันโดยปกติจะแสดงอยู่ที่เว็บ air4thai [air] ซึ่งเป็นเว็บไซต์แสดงค่าคุณภาพอากาศของกรมควบคุมมลพิษ แต่ในงานวิจัยนี้เนื่องจากเรามีการใช้ข้อมูลคุณภาพอากาศแบบย้อนหลัง เราจึงได้ทำคำร้องขอข้อมูลย้อนหลังโดยเฉพาะไปยังกรมควบคุมมลพิษโดยตรง โดยการวัดคุณภาพอากาศนั้นจะทำโดยการตั้งสถานีตรวจวัดไว้ตามจุดต่าง ๆ และค่าคุณภาพอากาศที่ได้นั้นจะขึ้นอยู่กับว่าสถานีตรวจวัดนั้น ๆ เจาะจงเอาค่ามลพิษตัวใดเป็นตัวแสดง เช่น PM 2.5, PM 10 หรือ NO2 เป็นต้น (โดยทั่วไปจะเจาะจงไปที่ค่า PM 2.5 หรือไม่ก็ PM 10 เป็นหลัก) โดยค่าคุณภาพอากาศจะเริ่มตั้งแต่ 0 ขึ้นไป และระดับคุณภาพอากาศจะแบ่งออกเป็น 5 ระดับ และแต่ละระดับจะมีผลกระทบต่อสุขภาพดังนี้

ค่าคุณภาพอากาศ	ระดับคุณภาพอากาศ
0 - 25	ดีมาก
26 – 50	ดี
51 – 100	ปานกลาง
101 – 200	เริ่มมีผลกระทบต่อสุขภาพ ผู้เป็นโรคระบบทางเดินหายใจควรหลีกเลี่ยง
201 ขึ้นไป	มีผลกระทบต่อสุขภาพ

ตารางที่ 2.3 ตารางแสดงระดับคุณภาพอากาศ ตามมาตรฐานของกรมควบคุมมลพิษ ฯ ประเทศไทย

2.4 การเรียนรู้ของเครื่อง (Machine Learning)

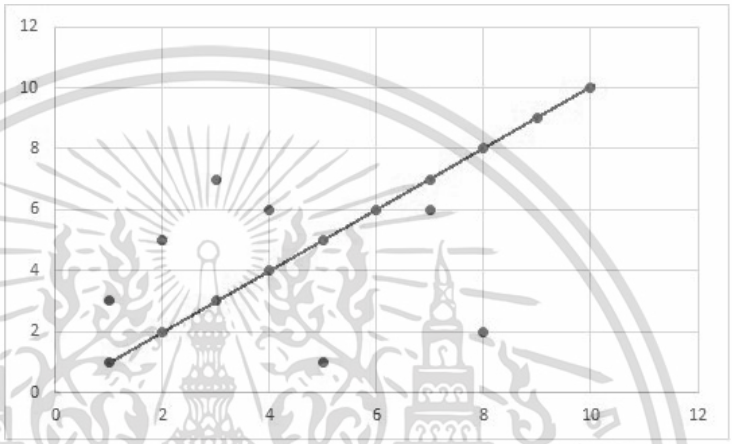
การเรียนรู้ของเครื่องที่ใช้ในวิทยานิพนธ์นี้เป็นการเรียนรู้ของเครื่องแบบมีผู้สอน (Supervised Machine Learning Techniques) กล่าวคือ จะมีข้อมูลชุดหนึ่งที่มีข้อมูลสำหรับฝึกสอนให้ตัวแบบที่จะใช้ทำนาย ประกอบไปด้วยชุดข้อมูลสำหรับเรียนรู้และชุดคำตอบ

2.4.1.ตัวแบบการเรียนรู้ของเครื่อง (Machine Learning Model)

ตัวแบบการเรียนรู้ของเครื่องที่จะทำการทดลองในวิทยานิพนธ์นี้จะใช้ตัวแบบจากเทคนิคย่อยของการเรียนรู้ของเครื่องแบบมีผู้สอน ซึ่งเทคนิคย่อยของการเรียนรู้แบบมีผู้สอนนั้นแยกได้ออกเป็น 3 เทคนิคคือ 1. Classification Techniques (คัดแยก) 2. Regression Techniques (วิเคราะห์ความสัมพันธ์เชิงเส้น คำตอบเป็นตัวเลข) และ 3. Reinforcement Techniques (ทดลองทำและนำความผิดพลาดจากการทดลองก่อนหน้ามาเป็นผู้สอน) ซึ่งในวิทยานี้จะใช้อยู่สองเทคนิคย่อย คือ ส่วนที่เป็นส่วนคัดแยกข้อมูลที่จะนำมาแยกข้อมูลนำเข้าว่าควรจะอยู่ในระดับคุณภาพอากาศแบบใด (Classifier Model) ซึ่งในวิทยานิพนธ์นี้ตัวแบบที่จะใช้ในการทดลองการคัดแยกข้อมูลคือ Decision Tree, Naïve Bayes, K Nearest Neighbors, Random Forest และ Gradient Boosting ส่วนที่สองเป็นส่วนของตัวแบบที่จะทำนายค่าคุณภาพอากาศออกมาเป็นตัวเลขอัน ซึ่งจะต้องใช้ตัวแบบถดถอยมาใช้ในการทำนาย (Regressor Model) ซึ่งตัวแบบการถดถอยที่จะนำมาใช้ในการทดลองในวิทยานิพนธ์นี้ได้แก่ Linear Regression, Decision Tree, K Nearest Neighbors, Random Forest และ Gradient Boosting

2.4.1.1 การวิเคราะห์การถดถอยเชิงเส้น (Linear Regression)

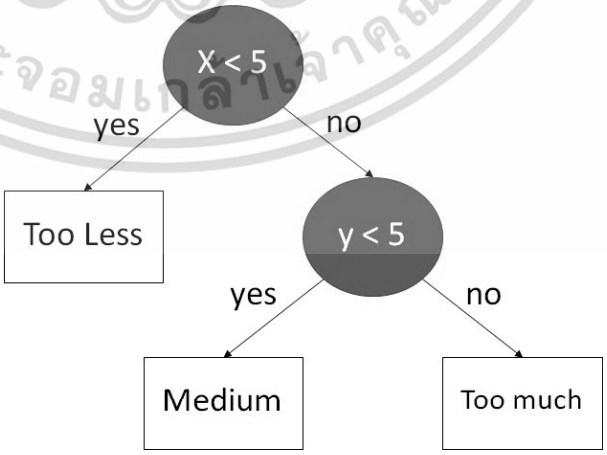
เป็นตัวแบบที่อยู่ในรูปสมการเชิงเส้น ($Y = A_1X_1+A_2X_2+A_3X_3+...+B$) โดยการปรับโครงสร้างของ Linear Regression จะทำโดยการค่าสัมประสิทธิ์ของทุกตัวแปร (A_1, A_2, A_3, ...) และค่าคงที่ (B) เพื่อให้เส้นตรงที่ได้มีความเหมาะสม (Fit)กับข้อมูลที่นำมาเรียนรู้มากที่สุด



รูปที่ 2.4 ตัวอย่างการวิเคราะห์เชิงเส้น โดยเส้นสีแดงคือตัวแบบ

2.4.1.2 ต้นไม้ตัดสินใจ (Decision Tree)

เป็นตัวแบบที่สร้างขึ้นจากโครงสร้างข้อมูลแบบต้นไม้ โดยที่โหนดภายใน (Internal Node) จะเป็นเงื่อนไขสำหรับการตัดสินใจ และโหนดใบ (Leaf Node) จะเป็นผลลัพธ์สำหรับการทำนาย



รูปที่ 2.5 รูปตัวอย่างต้นไม้ตัดสินใจ

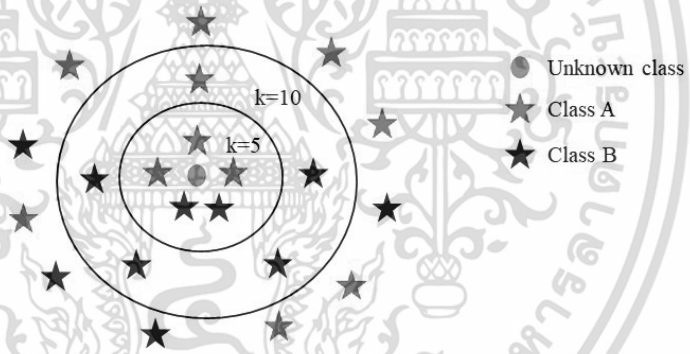
2.4.1.3 ตัวแบบจากกฎของเบย์อย่างง่าย (Naïve Bayes)

เป็นตัวแบบซึ่งใช้วิธีการทางสถิติโดยใช้ความน่าจะเป็น ซึ่งความน่าจะเป็นที่นำมาใช้ทำนายนั้น จะใช้กฎของเบย์ (Bayes Rule) โดยผลลัพธ์การทำนายจะมาจากการคำนวณความน่าจะเป็นของคำตอบจากทุก ๆ Features ของชุดข้อมูลสำหรับฝึกสอน

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

2.4.1.4 K Nearest Neighbors (KNN)

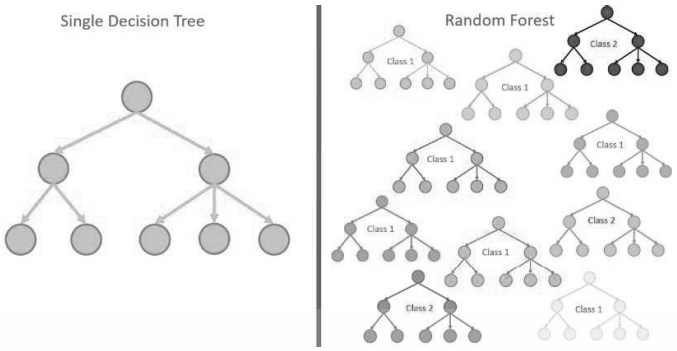
เป็นตัวแบบที่จะทำการทำนายตามผลลัพธ์ของชุดข้อมูลฝึกสอนที่อยู่ใกล้สุด ซึ่งระยะห่างระหว่างข้อมูล (Distance) สามารถคำนวณได้จากวิธีการต่าง ๆ โดยผู้ใช้มีหน้าที่ตั้งค่าจำนวนข้อมูลที่ใกล้ที่สุด (ค่า K) ที่จะนำมาใช่ร่วมทำนาย



รูปที่ 2.6 ตัวอย่างตัวแบบ K Nearest Neighbors

2.4.1.5 Random Forest [7]

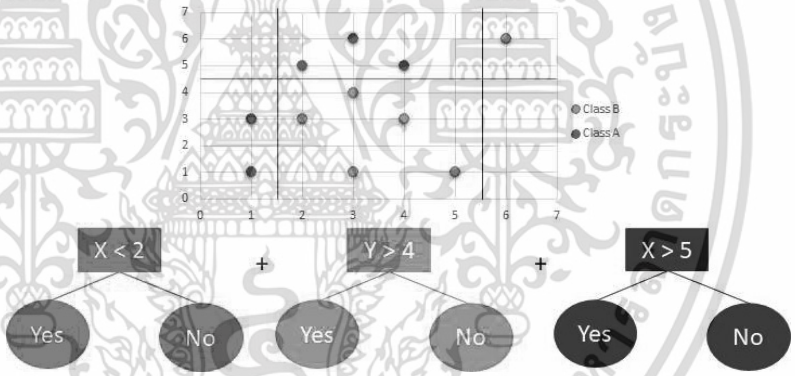
เป็นตัวแบบที่เกิดจากการสุ่มสร้างต้นไม้ตัดสินใจด้วยวิธีการต่าง ๆ กับชุดข้อมูลฝึกสอน โดย Random Forest เป็นหนึ่งในวิธีทางการเรียนรู้ของเครื่องที่เรียกว่า Ensemble Technique โดยผลลัพธ์การทำนายจะคำนวณจากผลลัพธ์การทำนายของต้นไม้ตัดสินใจแต่ละต้นที่ถูกสร้างขึ้น



รูปที่ 2.7 ด้านซ้ายต้นไม้ตัดสินใจเพียงต้นเดียว ด้านขวาตัวอย่าง Random Forest

2.4.1.6 Gradient Boosting [8]

เป็นตัวแบบอีกหนึ่งวิธีในการเรียนรู้ของเครื่องแบบ Ensemble Technique โดยเกิดจากการนำผลลัพธ์ที่ผิดพลาดของตัวแบบเก่ามาทำการทำนายอีกครั้งด้วยตัวแบบใหม่ จนกว่าจะได้ผลลัพธ์ที่พอใจ



รูปที่ 2.8 ตัวอย่างการทำงานของตัวแบบ Gradient Boosting

2.4.2 การวัดประสิทธิภาพของตัวแบบ

การที่จะตัดสินว่าตัวแบบการเรียนรู้ของเครื่องตัวใดมีประสิทธิภาพกว่ากัน หรือเหมาะสมกับการใช้งานในวิทยานิพนธ์นี้ ย่อมต้องมีวิธีการวัดประสิทธิภาพการทำงานของแต่ละตัวแบบด้วยวิธีต่าง ๆ เนื่องจากวิทยานิพนธ์นี้เราได้มีการใช้ทั้งตัวแบบสำหรับคัดแยกข้อมูล (Classifier Model) และตัวแบบถดถอย (Regressor Model) ซึ่งตัวแบบแต่ละประเภทมีการวิเคราะห์ประสิทธิภาพต่างกันดังนี้

2.4.2.1 การวัดประสิทธิภาพของตัวแบบคัดแยกข้อมูล (Classifier Model)

2.4.2.1.1 Confusion Matrix [9]

เป็นการจำแนกผลลัพธ์การทำนายของตัวแบบ โดยทั่วไปมักจะพบเห็น Confusion Matrix ของผลลัพธ์การทำนายแบบสองผลลัพธ์ เช่น ใช่-ไม่ใช่ ชาย-หญิง แต่เนื่องจากวิทยานิพนธ์นี้ตัวแบบจะต้องแยกผลลัพธ์การทำนายออกเป็น 5 ผลลัพธ์ตามจำนวนคุณภาพอากาศ จึงจะต้องมีการคำนวณผลลัพธ์ของ Confusion Matrix ดังนี้

		Actual				
	Class	1	2	3	4	5
Predict	1	A	B	C	D	E
	2	F	G	H	I	J
	3	K	L	M	N	O
	4	P	Q	R	S	T
	5	U	V	W	X	Y

ตารางที่ 2.4 ตัวอย่าง Confusion Matrix แบบ 5 ผลลัพธ์

ตัวอย่างการคำนวณ จะยกตัวอย่างเจาะจงไปที่ผลลัพธ์การทำนาย = 1

$$TP_1 = A$$

$$TN_1 = G + M + S + Y$$

$$FP_1 = B + C + D + E$$

$$FN_1 = F + K + P + U$$

2.4.2.1.2 Accuracy

หมายถึงความแม่นยำรวมของตัวแบบ

$$Accarucy = \frac{TP}{N}$$

เมื่อ TP คือ ผลลัพธ์ที่ทำนายถูกทั้งหมด (A + G + M + S + Y)

N คือ จำนวนข้อมูลทั้งหมด

2.4.2.1.3 Precision

หมายถึงอัตราส่วน การทำนายเป็น class นั้น ๆ ถูก ต่อจำนวนที่ข้อมูลที่ตัวแบบทำนายเป็น class นั้น ๆ

$$Precision_x = \frac{TP_x}{TP_x + FP_x}$$

เมื่อ x คือ class ที่สนใจ

TP คือ ผลลัพธ์ที่ทำนายถูกใน class ที่สนใจ

FP คือ ผลลัพธ์ที่ทำนายเป็น class ที่สนใจ แต่เป็นผลการทำนายที่ผิด

2.4.2.1.4 Recall

หมายถึงอัตราส่วน การทำนายเป็น class นั้น ๆ ถูก ต่อจำนวนข้อมูลของ class นั้น ๆ

$$Recall_x = \frac{TP_x}{TP_x + FN_x}$$

เมื่อ x คือ class ที่สนใจ

TP คือ ผลลัพธ์ที่ทำนายถูกใน class ที่สนใจ

FN คือ ผลลัพธ์ที่ทำนายเป็น class อื่น ๆ แต่ความเป็นจริงต้องเป็น class ที่สนใจ

2.4.2.1.5 F1 Score

หมายถึง อัตราส่วนเฉลี่ยระหว่าง Precision และ Recall

$$F1\ Score_x = 2\frac{Precision_x * Recall_x}{Precision_x + Recall_x}$$

เมื่อ x คือ class ที่สนใจ

2.4.2.2 การวัดประสิทธิภาพของตัวแบบความถดถอย (Regressor Model)

2.4.2.2.1 การวิเคราะห์ความถดถอย (Coefficient of Determination - R^2) [10]

R^2 เป็นการวิเคราะห์ว่า ตัวแบบสำหรับทำนาย ที่ได้นั้นเหมาะสมกับชุดข้อมูลนี้หรือไม่ กล่าวคือ เส้นกราฟที่ได้จาก ตัวแบบสำหรับทำนาย มีความเหมาะสม (Fit) กับข้อมูลหรือไม่ หรืออีกนัยหนึ่ง R^2 เป็นการวิเคราะห์ถึงความเหมาะสมของสัมประสิทธิ์ที่ได้จาก ตัวแบบสำหรับทำนาย นั่นเอง

$$R^2 = 1 - \frac{\sum_i (predict_i - observ_i)^2}{\sum_i (predict_i - \overline{predict})^2}$$

2.4.2.2.2 Mean Absolute Error (MAE)

MAE เป็นการวิเคราะห์ค่าความคลาดเคลื่อนของค่าทำนายของตัวแบบเมื่อเปรียบเทียบกับข้อมูลจริง โดยค่าที่นำมาวิเคราะห์จะเป็นค่าเฉลี่ยของค่าสัมบูรณ์จากผลต่างของค่าทำนายและข้อมูลจริง

$$MAE = \frac{\sum_i |predict_i - observ_i|}{n}$$

2.5 งานวิจัยที่เกี่ยวข้อง

ในปีค.ศ. 2012 Chitini และคณะ [11] และคณะ ได้เสนอวิธีการวัดค่ามลพิษโดยการใช้ภาพถ่ายดาวเทียมของดาวเทียม IRS เปรียบเทียบกับดาวเทียม Landsat 7 โดยในงานวิจัยมีการคำนวณค่า Vegetable Indices – VI, NDVI และ TVI มาใช้ในการคำนวณร่วมกับภาพถ่ายดาวเทียม โดยได้มีการแสดงค่า R^2 ของค่าต่าง ๆ เพื่อแสดงถึงความสัมพันธ์เชิงเส้นของค่าแต่ละตัวในชุดข้อมูลกับค่าคุณภาพอากาศ และสุดท้ายได้แสดงสมการเชิงเส้นสำหรับคำนวณค่าคุณภาพอากาศของแต่ละภาพถ่ายดาวเทียมที่นำมาใช้ในงานของตน

ในปีค.ศ. 2016 Qian di และคณะ [12] ได้เสนอวิธีการใช้ Neural Networks มาทำนายค่าคุณภาพอากาศเจาะจงไปที่ค่ามลพิษ PM 2.5 โดยใช้ข้อมูล Aerosol Optical Depth (AOD) ร่วมกับข้อมูลเชิงพื้นที่อื่น ๆ

ในปีค.ศ. 2017 Husabir และคณะ [13] ได้เสนอวิธีการทาง SWARM Optimization อย่าง Particles SWARM Optimization (PSO) เพื่อทำนายค่า Benzene ในอากาศจากข้อมูลเชิงพื้นที่ต่าง ๆ ซึ่งค่า Benzene ถือเป็นหนึ่งในมลพิษทางอากาศ

ในปีค.ศ. 2019 Mehdi และคณะ[14] ได้ใช้ข้อมูลเชิงพื้นที่ของเมืองเตหะราน ประเทศอิหร่าน มาทำนายค่า PM 2.5 ด้วยตัวแบบทางการเรียนรู้ของเครื่อง ได้แก่ Ransom Forest, Extreme Gradient Boosting (XGBoost) และได้มีการเรียนรู้เชิงลึก (Deep Learning) เข้ามาร่วมทำงานในงานนี้ด้วย งานวิจัยของ Jasleen และ Mamta [15] ได้เสนอการใช้ข้อมูลเชิงพื้นที่ของเมืองฟาริดาบาด ประเทศอินเดีย มาวิเคราะห์คุณภาพอากาศโดยใช้วิธีการทางการเรียนรู้ของเครื่อง และยังมีงานวิจัยของ Weilin และคณะ [16] ที่ใช้การเรียนรู้เชิงลึก (Deep Learning) มาทำนายค่า PM 2.5 ในประเทศจีนโดยใช้ข้อมูลเชิงพื้นที่อีกด้วย

ในปีค.ศ. 2020 Xiankun Sun และคณะ [17] ได้ใช้ข้อมูล MODIS ซึ่งเป็นข้อมูลชนิดหนึ่งซึ่งสามารถเก็บได้จากดาวเทียม ร่วมกับข้อมูลเชิงพื้นที่อื่น ๆ มาใช้พยากรณ์หมอกมลพิษทางอากาศ และยังมีงานวิจัยของ W.C. Leong และคณะ [18] ที่ได้มีการใช้ Supported Vector Machine หรือที่รู้จักกันในนามของ SVM ซึ่งเป็นอีกหนึ่งวิธีในการเรียนรู้ของเครื่อง มาใช้ทำนายค่าคุณภาพอากาศ

ในปีค.ศ. 2022 Yuanlin Gu และคณะ [19] ได้เสนอวิธีการสร้างตัวแบบ Hybrid Interpretable Artificial Neural Network มาทำการทำนายค่าคุณภาพอากาศโดยใช้ข้อมูลเชิงพื้นที่

งานวิจัยของ Fangzhou Lin และคณะ [20] ได้ใช้ Convolution Neural Networks (CNNs) มาทำการตรวจสอบการเคลื่อนที่ของมลพิษในอากาศผ่านทางดาวเทียม Himawari-8 (ปัจจุบัน Himawari-8 เป็นหนึ่งในข้อมูลดาวเทียมที่กรมอุตุนิยมวิทยา ประเทศไทย ได้ใช้ในการทำนายสภาพอากาศรายวัน) งานวิจัยของ Huimin Ji และคณะ [21] ได้มีการเสนอวิธีการตรวจสอบคุณภาพอากาศ โดยใช้ข้อมูลสุขภาพที่มีเก็บได้จาก Weibo ซึ่งเป็น Social Media ที่ได้รับความนิยมมากในประเทศจีน งานวิจัยของ Marc Saez และ Maria A. Barcelo [22] ได้เสนอการใช้วิธีการ Hierarchical Bayesian spatiotemporal ในการทำนายคุณภาพอากาศของแคว้นกาตาลัน ประเทศสเปน และยังมีงานของ Andrew A. Boateng [23] ที่เสนอการทำงานร่วมกันของตัวแบบคัดแยกและตัวแบบถดถอย ซึ่งถือเป็นวิธีการที่นำมาใช้แก้ปัญหาในวิทยานิพนธ์นี้

ข้อดี

1. แต่ละงานวิจัยแสดงวิธีการใช้ข้อมูลเชิงพื้นที่ ซึ่งหากนำไปใช้จริงสามารถเพิ่มความสามารถในการตรวจวัดค่ามลพิษทางอากาศ โดยไม่จำเป็นต้องตั้งสถานีตรวจวัดเพิ่ม
2. แต่ละงานวิจัยมีการประยุกต์ใช้การเรียนรู้ของเครื่องในวิธีการต่าง ๆ เพื่อเพิ่มประสิทธิภาพในการทำนายผล
3. บางงานวิจัยมีการใช้ข้อมูลภาพถ่ายดาวเทียมซึ่งเป็นข้อมูลที่เกิดจากการรับคลื่นที่สะท้อนมาจากโลก ซึ่งน่าจะทำการหาผลลัพธ์ของค่าคุณภาพอากาศผ่านข้อมูลเหล่านั้นได้

ข้อเสีย

1. งานวิจัยของ Chitini และคณะ ถึงแม้จะใช้เพียงข้อมูลภาพถ่ายดาวเทียมเพียงอย่างเดียว แต่ยังมีจุดที่ยังเพิ่มเติมได้คือ ในงานนั้นยังไม่มีการใช้การเรียนรู้ของเครื่องมาเปรียบเทียบ
2. ส่วนงานอื่น ๆ ที่มีการใช้การเรียนรู้ของเครื่อง ก็ยังมีข้อเสียอยู่เช่น การใช้ข้อมูลจากหลายแหล่งซึ่งบางครั้งอาจจะทำให้ยากต่อการรวบรวม
3. บางงานใช้ภาพถ่ายดาวเทียมแต่นำไปใช้กับวิธีการเรียนรู้เชิงลึก แต่ไม่ได้แสดงผลลัพธ์ของการใช้การเรียนรู้ของเครื่องแบบปกติ ซึ่งการเรียนรู้เชิงลึกอาจจะให้ผลลัพธ์ที่ดี แต่จะเปลืองทรัพยากรในการประมวลผลที่มาก

บทที่ 3
วิธีการดำเนินงานวิจัย

3.1 หาข้อมูลภาพถ่ายดาวเทียมซึ่งในงานวิจัยนี้จะใช้ภาพถ่ายจากดาวเทียม Landsat-8 ซึ่งข้อมูลภาพถ่ายดาวเทียมที่นำมาใช้จะเป็นข้อมูล Level 2 (SR) ข้อมูลสามารถสั่งซื้อฟรีได้ผ่านเว็ปไซต์ https://earthexplorer.usgs.gov/ โดยจะต้องสมัครสมาชิกกับทางเว็ปไซต์ก่อน

3.2 หาข้อมูลคุณภาพอากาศจากเว็บไซต์ www.air4thai.pcd.go.th ในวันและเวลาที่ตรงกับการถ่ายภาพจากดาวเทียม (Landsat-8 จะมีการถ่ายภาพ ณ จุดเดิม ทุก 16 วัน) โดยข้อมูลภาพถ่ายดาวเทียมที่นำมาใช้มีรายละเอียดดังนี้

วันที่	Scene	ความละเอียด (Pixel)	ขนาด	ภาพเสียหาย
20 พฤษภาคม 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
20 พฤษภาคม 2561	129051	7721 * 7551	1.03 GB	เสียหายเล็กน้อย
5 มิถุนายน 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
5 มิถุนายน 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
21 มิถุนายน 2561	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
21 มิถุนายน 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
7 กรกฎาคม 2561	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
7 กรกฎาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
23 กรกฎาคม 2561	129050	7721 * 7551	1.03 GB	เสียหายเล็กน้อย
23 กรกฎาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
8 สิงหาคม 2561	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
8 สิงหาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
24 สิงหาคม 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
24 สิงหาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
9 กันยายน 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
9 กันยายน 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
25 กันยายน 2561	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
25 กันยายน 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
11 ตุลาคม 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
11 ตุลาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
27 ตุลาคม 2561	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย

วันที่	Scene	ความละเอียด	ขนาด	ภาพเสียหาย
27 ตุลาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
12 พฤศจิกายน 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
12 พฤศจิกายน 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
28 พฤศจิกายน 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
28 พฤศจิกายน 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
14 ธันวาคม 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
14 ธันวาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
30 ธันวาคม 2561	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
30 ธันวาคม 2561	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
15 มกราคม 2562	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
15 มกราคม 2562	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
31 มกราคม 2562	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
31 มกราคม 2562	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
16 กุมภาพันธ์ 2562	129050	7711 * 7551	1.03 GB	เสียหายเล็กน้อย
16 กุมภาพันธ์ 2562	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
4 มีนาคม 2562	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
4 มีนาคม 2562	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
20 มีนาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
20 มีนาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
5 เมษายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
5 เมษายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
21 เมษายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
21 เมษายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
7 พฤษภาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
7 พฤษภาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
23 พฤษภาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
23 พฤษภาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
8 มิถุนายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
8 มิถุนายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
24 มิถุนายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
24 มิถุนายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย

วันที่	Scene	ความละเอียด	ขนาด	ภาพเสียหาย
10 กรกฎาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
10 กรกฎาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
26 กรกฎาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
26 กรกฎาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
11 สิงหาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
11 สิงหาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
27 สิงหาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
27 สิงหาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
12 กันยายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
12 กันยายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
28 กันยายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
28 กันยายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
14 ตุลาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
14 ตุลาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
30 ตุลาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
30 ตุลาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
15 พฤศจิกายน 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
15 พฤศจิกายน 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
1 ธันวาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
1 ธันวาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
17 ธันวาคม 2562*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
17 ธันวาคม 2562*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
2 มกราคม 2563*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
2 มกราคม 2563*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
18 มกราคม 2563*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
18 มกราคม 2563*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
3 กุมภาพันธ์ 2563*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
3 กุมภาพันธ์ 2563*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย
19 กุมภาพันธ์ 2563*	129050	7711 * 7561	1.03 GB	เสียหายเล็กน้อย
19 กุมภาพันธ์ 2563*	129051	7721 * 7561	1.03 GB	เสียหายเล็กน้อย

ตารางที่ 3.1 ตารางแสดงถึงวันและเวลาของภาพถ่ายดาวเทียมที่นำมาใช้

* เป็นข้อมูลที่เพิ่มมาภายหลังจากการทดลองครั้งแรก โดยจะกล่าวในภายหลัง

3.3 จัดเตรียมข้อมูลเพื่อให้พร้อมใช้งานสำหรับงานวิจัยนี้ เช่น จัดเก็บลงฐานข้อมูล ติดตั้งเครื่องมือที่จะใช้ร่วมกับข้อมูล ทำการคลีนและคำนวณค่าต่าง ๆ ใส่ไปยังชุดข้อมูล เป็นต้น ซึ่งวิธีการทำงานกับข้อมูลมีดังนี้

3.3.1 ฐานข้อมูลที่นำมาใช้คือ PostgreSQL ซึ่งเป็นฐานข้อมูลที่ใช้แพร่หลายในการจัดเก็บข้อมูลเชิงพื้นที่

3.3.2 Open Data Cube API Platform สำหรับใช้ร่วมกับฐานข้อมูลสำหรับ นำข้อมูลลงฐานข้อมูล การเรียกข้อมูลขึ้นมาใช้งาน รวมถึงการทำงานบางอย่าง โดย Open Data Cube เป็น API Library ที่พัฒนาขึ้นบนภาษา Python

3.3.3 ทำการ Query ข้อมูลที่มีจุดตรงกับสถานีตรวจวัด และทำการนำข้อมูล AQI ของสถานีตรวจวัดเข้ามาประกอบในชุดข้อมูลโดยตำแหน่งพิกัดทางภูมิศาสตร์ตรงกับตำแหน่งของสถานีตรวจวัด โดยข้อมูลพิกัดทางภูมิศาสตร์ (SRID EPSG 4326 : WGS84)ได้มาจาก www.air4thai.pcd.go.th ดังนี้

สถานี	แขวง/ตำบล	เขต/อำเภอ	จังหวัด	Latitude	Longitude
02t	หิรัญรูจี	ธนบุรี	กทม.	13.727557	100.486604
03t	แสมดำ	บางขุนเทียน	กทม.	13.636514	100.414262
05t	บางนา	บางนา	กทม.	13.666113	100.605741
08t	ทรงคะนอง	พระประแดง	สมุทรปราการ	13.664023	100.543406
10t	คลองจั่น	บางกะปิ	กทม.	13.779539	100.645654
11t	ดินแดง	ดินแดง	กทม.	13.775516	100.569206
12t*	ช่องนนทรี	ยานนาวา	กทม.	13.708038	100.54735
13t*	ตลาดขวัญ	เมือง	นนทบุรี	13.807156	100.50632
14t*	อ้อมน้อย	กระทุ่มแบน	สมุทรสาคร	13.705458	100.31568
17t*	ตลาด	พระประแดง	สมุทรปราการ	13.652154	100.53184
18t*	ปากน้ำ	เมือง	สมุทรปราการ	13.599172	100.59733
19t*	บางเสาธง	บางเสาธง	สมุทรปราการ	13.570333	100.78587
20t*	คลองหนึ่ง	คลองหลวง	ปทุมธานี	14.037512	100.60512
22t*	บางพูด	ปากเกร็ด	นนทบุรี	13.90794	100.5356

สถานี	แขวง/ตำบล	เขต/อำเภอ	จังหวัด	Latitude	Longitude
27t*	มหาชัย	เมือง	สมุทรสาคร	13.550478	100.26425
50t*	ปทุมวัน	ปทุมวัน	กทม.	13.729984	100.53644
52t*	ธนบุรี	ธนบุรี	กทม.	13.727557	100.4866
53t*	วังทองหลาง	วังทองหลาง	กทม.	13.795414	100.5929
54t*	ดินแดง	ดินแดง	กทม.	13.762609	100.55036
59t*	พญาไท	พญาไท	กทม.	13.783143	100.54053
61t*	พลับพลา	วังทองหลาง	กทม.	13.76963	100.61456

ตารางที่ 3.2 ตารางแสดงตำแหน่งที่ตั้งของสถานีตรวจวัดคุณภาพอากาศที่ใช้ในงานวิจัย [24]

* เป็นข้อมูลที่เพิ่มมาภายหลังจากการทดลองครั้งแรก โดยจะกล่าวในภายหลัง

รูปที่ 3.1 ภาพแสดงจุดที่ตั้งของสถานีตรวจวัดคุณภาพอากาศในกรุงเทพและปริมณฑล

3.3.4 แปลงรูปแบบของข้อมูลให้อยู่ในรูปของ Pandas Dataframe (เป็นไลบรารี่ในภาษา Python) เพื่อให้ง่ายต่อการตรวจสอบ และคำนวณในขั้นถัดไป

3.3.5 เนื่องจากค่าของข้อมูลที่จัดเก็บจะอยู่ในรูปของจำนวนเต็ม แต่การใช้งานค่าของข้อมูลโดยทั่วไปจะใช้อยู่ที่ 0 – 1 จึงต้องทำการแปลงค่าของข้อมูลโดยทำการหารค่าทุกตัวในชุดข้อมูลด้วย 10000

3.3.5 ทำการดรอปข้อมูลที่มีค่าที่เป็นค่ารบกวนออก เช่น ค่าที่ได้จาก band มีค่าคิดลบ หรือมีค่ามากกว่า 1

3.3.6 ทำการตัดข้อมูลที่มีการรบกวนของเมฆในระดับปานกลางถึงสูงออก ซึ่งสามารถดูได้จากค่า Pixel_QA ที่อยู่ในชุดข้อมูล ซึ่งค่า Pixel_QA จะอยู่ในรูปของจำนวนเต็ม ซึ่งเราจะต้องแปลงค่านั้นออกเป็นค่าเลขฐานสองจำนวน 16 bit ก่อนซึ่งแต่ละ bit ก็จะมีความหมายตามที่มีไว้ใน Guidebook ของ Landsat 8

3.3.7 ทำการคำนวณค่า Vegetable Indices แล้วนำเข้าประกอบกับชุดข้อมูล

3.3.8 ทำการจัดเก็บข้อมูลลงไฟล์ชนิดในชนิดหนึ่งเพื่อให้ง่ายต่อการเรียกใช้งานในรอบถัดไป ในที่นี้จะทำการบันทึกไฟล์เป็นนามสกุล csv เพื่อสามารถนำข้อมูลมาตรวจสอบด้วยตาเปล่าได้

3.4 จัดเตรียมไลบรารี่สำหรับการเรียนรู้ของเครื่องซึ่งในที่นี้เราจะใช้ Scikitlearn รวมถึงไลบรารี่อื่น ๆ ที่เกี่ยวข้อง

3.5 สร้างและปรับโครงสร้างของตัวแบบสำหรับทำนาย และทำการฝึกสอนตัวแบบด้วยข้อมูลที่จัดเตรียมไว้ ในวิทยานิพนธ์นี้จะมีการจัดสรรข้อมูลสำหรับเรียนรู้ออกเป็นสองชุดในลักษณะ 70/30 (แบ่งของมูลเป็น 70% และ 30% ของชุดข้อมูลที่มีทั้งหมด) ซึ่งข้อมูลในส่วน 70 เปอร์เซ็นต์แรกจะนำไปใช้ในการฝึกสอนตัวแบบสำหรับทำนาย (Trainset) ส่วนอีก 30 เปอร์เซ็นต์ที่เหลือจะใช้ในการทดลองทำนายเพื่อประเมินประสิทธิภาพของตัวแบบ (Testset / Validateset) โดยการแบ่งข้อมูลนั้นจะทำการสุ่มแบ่งลงไปในแต่ละระดับคุณภาพของอากาศในลักษณะ 70/30 และจะทำการเก็บชุดข้อมูลที่แบ่งไว้เพื่อใช้ในการฝึกสอนทุกตัวแบบที่จะทำการทดลอง

3.6 วัดและประเมินผล ผลลัพธ์ที่ได้จากตัวแบบที่สร้างขึ้น โดยจะมีการวัดผลกับข้อมูลในชุดข้อมูลทดสอบ (ที่แบ่งไว้ 30% จากข้อ 3.5)

3.7 จดบันทึก อภิปราย ชี้จุดที่สามารถพัฒนาได้เพิ่มเติมและสรุปผลที่ได้จากงานวิจัย

บทที่ 4

ผลการทดลอง

4.1 ในขั้นต้นเราได้ทำการทดลองสร้างตัวแบบสำหรับทำนาย จากตัวแบบถดถอย (Regressor Model) เพียงอย่างเดียว โดยได้ทำการทดลองสร้างจาก Linear Regression และได้มีการทดลองสร้าง Artificial Neural Network (ANN) ขึ้นมาร่วมทำนายได้ แต่ได้ผลลัพธ์ที่ยังไม่น่าพอใจ กล่าวคือ ได้ผลลัพธ์เป็นค่า Coefficient of Determination หรือ R^2 น้อยกว่า 0.1 ซึ่งแปลว่าข้อมูลที่นำมาใช้นั้นไม่มีความสัมพันธ์เชิงเส้นกับค่าคุณภาพอากาศ และได้ค่า Mean Absolute Error อยู่ที่ประมาณ 30 ซึ่งเกินช่วงที่สั้นที่สุดของระดับคุณภาพอากาศ ขณะนั้นเรามีจำนวนในชุดข้อมูลอยู่ที่ประมาณ 390 ชุดข้อมูล ซึ่งน้อยมากหากจะใช้วิธีทาง Deep learning

4.2 เราจึงได้ทดสอบสร้าง Model สำหรับคัดแยกข้อมูลมาตรวจสอบว่าจะมีความเป็นไปได้หรือไม่ที่เราจะหาคุณภาพอากาศจากภาพถ่ายดาวเทียมผ่านการเรียนรู้ของเครื่อง โดยตัวแบบที่เราได้เลือกมาใช้ ได้แก่

4.2.1 Decision Tree

4.2.2 Naïve Bayes

4.2.3 K Nearest Neighbors (KNN)

4.2.4 Random Forest

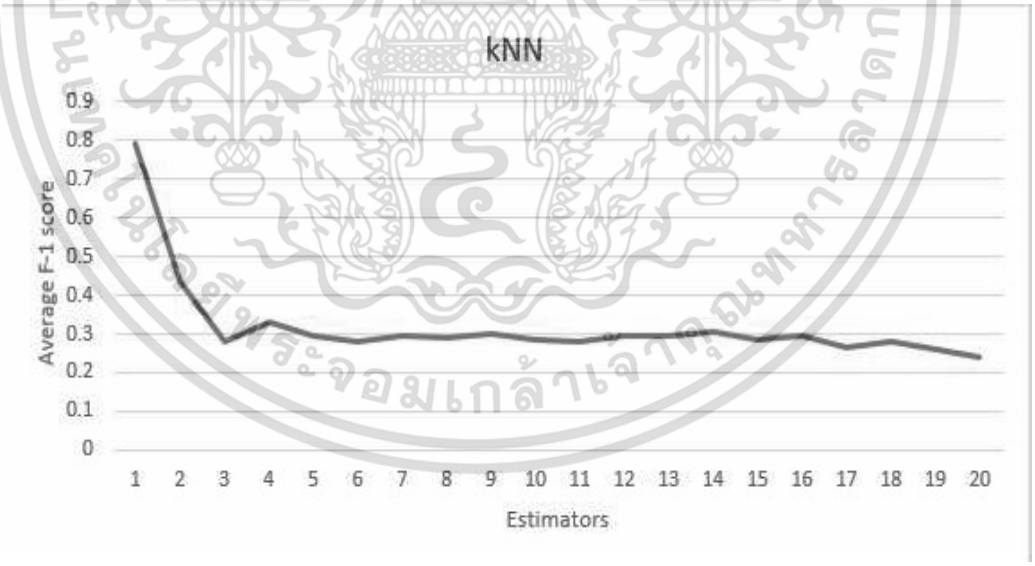
4.2.5 Gradient Boosting

โดยการปรับแต่งตัวแบบสำหรับทำนายมีดังนี้

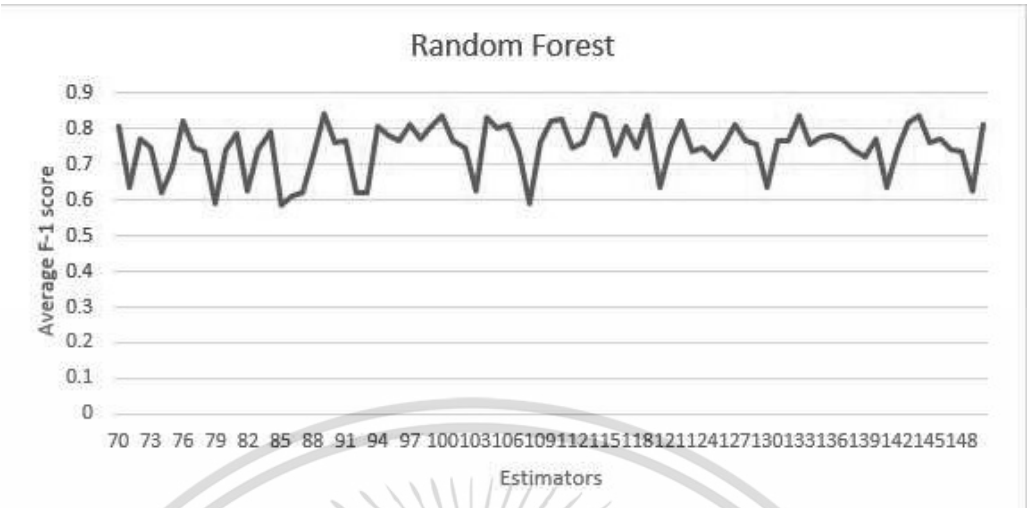
Model	Test Range
Decision Tree	Built with entropy theory.
Naïve Bayes	Built with Bayes's theory
K Nearest Neighbors (KNN)	1 to 20
Random Forest	70 to 200
Gradient Boosting	70 to 200

ตารางที่ 4.1 ตารางแสดงวิธีการปรับโครงสร้างของตัวแบบสำหรับทำนาย

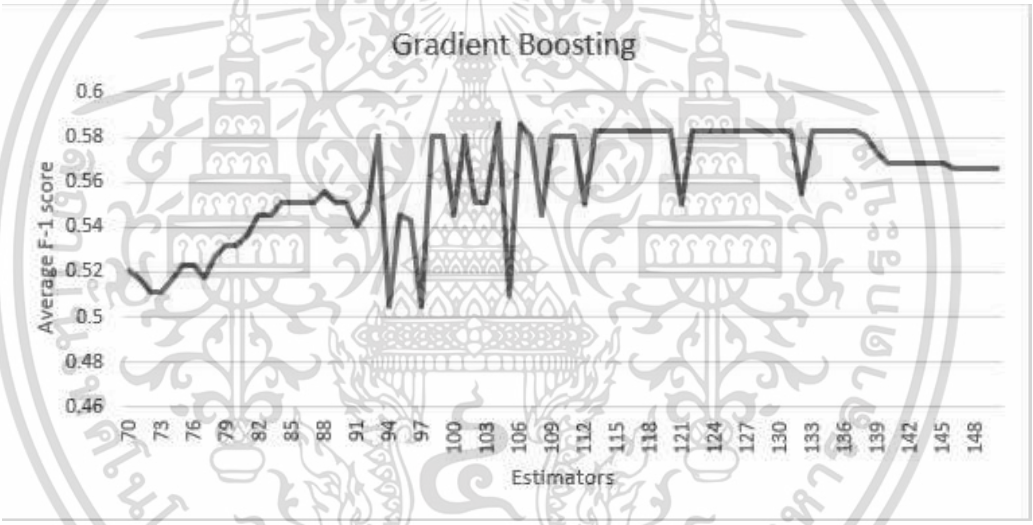
จะเห็นได้ว่าในส่วนของ K Nearest Neighbors (KNN), Random Forest และ Gradient Boosting นั้น ยังต้องมีการปรับแต่งค่า k หรือจำนวนตัวแบบสำหรับทำนาย (Random Forest และ Gradient Boosting เป็น Ensemble Techniques หมายถึงการใช้ตัวแบบสำหรับทำนายหลาย ๆ ตัวมาทำงานร่วมกันด้วยวิธีการต่าง ๆ) เราจึงต้องทำการหาค่า k หรือ จำนวนตัวแบบสำหรับทำนายที่เหมาะสมที่สุดก่อน ซึ่งได้ผลลัพธ์ดังนี้



รูปที่ 4.1 กราฟแสดงผลลัพธ์การหาค่า k ที่เหมาะสมที่สุดของตัวแบบ KNN



รูปที่ 4.2 กราฟแสดงผลลัพธ์การหาจำนวนตัวแบบที่เหมาะสมที่สุดของตัวแบบ Random Forest



รูปที่ 4.3 กราฟแสดงผลลัพธ์การหาจำนวนตัวแบบที่เหมาะสมที่สุดของตัวแบบ Gradient Boosting

Model	Loss function	K or N Estimators
Decision Tree	Entropy	-
Naïve Bayes	Bayes theory	-
K Nearest Neighbors (KNN)	Distance	1
Random Forest	Entropy	114
Gradient Boosting	Entropy	106

ตารางที่ 4.2 ตารางแสดงโครงสร้างของตัวแบบสำหรับทำนาย

ซึ่งได้ผลลัพธ์ของการทดลองดังนี้

Class	Very Good	Good	Satisfactory	Unhealthy	Very Unhealthy
Accuracy	0.78	0.79	0.94	0.93	0.99
Precision	0.77	0.69	0.6	0.88	0.5
Recall	0.73	0.81	0.5	0.78	1
F1 - Score	0.7333	0.7452	0.5455	0.8248	0.6667

ตารางที่ 4.3 ตารางแสดงผลลัพธ์ของตัวแบบ Decision Tree

Class	Very Good	Good	Satisfactory	Unhealthy	Very Unhealthy
Accuracy	0.41	0.46	0.85	0.46	0.72
Precision	0.46	0	0	0.25	0.07
Recall	0.51	0	0	0.56	1
F1 - Score	0.4837	0	0	0.3457	0.1308

ตารางที่ 4.4 ตารางแสดงผลลัพธ์ของตัวแบบ Naïve Bayes

Class	Very Good	Good	Satisfactory	Unhealthy	Very Unhealthy
Accuracy	0.85	0.89	0.97	0.94	0.99
Precision	0.89	0.9	0.8	0.8	0.5
Recall	0.9	0.78	0.67	0.89	1
F1 - Score	0.8764	0.8358	0.7273	0.8421	0.6667

ตารางที่ 4.5 ตารางแสดงผลลัพธ์ของตัวแบบ KNN

Class	Very Good	Good	Satisfactory	Unhealthy	Very Unhealthy
Accuracy	0.83	0.82	0.98	0.94	1
Precision	0.78	0.74	1	0.93	1
Recall	0.84	0.78	0.67	0.78	1
F1 - Score	0.809	0.7568	0.8	0.8485	1

ตารางที่ 4.6 ตารางแสดงผลลัพธ์ของตัวแบบ Random Forest

Class	Very Good	Good	Satisfactory	Unhealthy	Very Unhealthy
Accuracy	0.82	0.82	0.95	0.91	0.99
Precision	0.8	0.76	0.75	0.74	0
Recall	0.81	0.78	0.5	0.78	0
F1 - Score	0.8046	0.7671	0.6	0.7568	0

ตารางที่ 4.7 ตารางแสดงผลลัพธ์ของตัวแบบ Gradient Boosting

Model	Average Accuracy*	Average Precision*	Average Recall*	Average F1 Score*
Decision Tree	0.886	0.688	0.758	0.70308
Naïve Bayes	0.58	0.156	0.414	0.19205
K Nearest Neighbors (KNN)	0.936	0.77	0.848	0.78965
Random Forest	0.914	0.89	0.814	0.84285
Gradient Boosting	0.898	0.61	0.574	0.5857

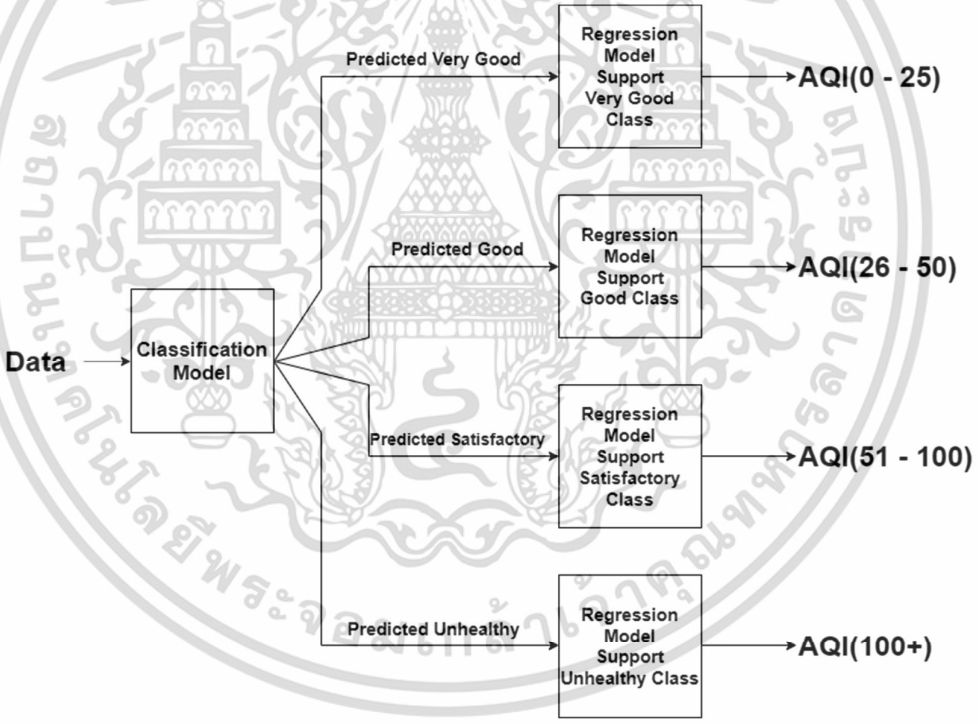
ตารางที่ 4.8 ตารางแสดงการเปรียบเทียบผลลัพธ์ของตัวแบบสำหรับทำนาย

* ค่าเฉลี่ยที่แสดงเป็นค่าเฉลี่ยแบบ Macro Average กล่าวคือเป็นค่าเฉลี่ยโดยไม่สนจำนวนของข้อมูล

จากผลลัพธ์ที่ได้จะเห็นได้ว่าตัวแบบ KNN ให้ผลลัพธ์ในด้าน Accuracy ดีที่สุดและตัวแบบ Random Forest ให้ค่า Accuracy ดีที่สุดลำดับที่สอง ในทางกลับกัน ตัวแบบ Random Forest ให้ค่า F1 Score ดีที่สุด ตัวแบบ KNN ให้ผลลัพธ์ในด้าน F1 Score ดีที่สุดลำดับที่สอง จากผลลัพธ์ทั้งหมดจะเห็นได้ว่า ความแตกต่างของค่า Accuracy ของลำดับที่หนึ่งและลำดับที่สองนั้นมีความแตกต่างกัน

น้อยมาก แต่ในทางกลับกันความแตกต่างของค่า F1 Score ของลำดับที่หนึ่งและลำดับที่สองนั้นมีความแตกต่างกันมาก จึงสรุปได้ว่าตัวแบบ Random Forest ให้ผลลัพธ์ดีที่สุด ดังที่ได้แสดงไว้ในเอกสารประกอบงานประชุมวิชาการ ECTI-CON [conf]

4.3 เมื่อเราได้ทำการทดลองสร้างตัวแบบสำหรับคัดแยกมาทดลองทำนายระดับคุณภาพอากาศแล้วได้ผลลัพธ์ที่ดี ขั้นต่อมาเราจึงได้กลับไปยังจุดมุ่งหมายเดิมของเราคือต้องการสร้างตัวแบบเพื่อทำนายค่าคุณภาพมลพิษ โดยเราจะใช้ตัวแบบคัดแยก เป็นการแบ่งข้อมูลลงไปก่อนว่าน่าจะอยู่ในระดับคุณภาพอากาศใด จากนั้นจึงนำข้อมูลนั้นเข้าตัวแบบสำหรับทำนายค่าคุณภาพอากาศ ซึ่งเราจะมีตัวแบบถดถอยสำหรับทำนายค่าคุณภาพอากาศในแต่ละระดับคุณภาพอากาศที่ข้อมูลนำเข้าได้รับการทำนายมา โดยเราจะเรียกตัวแบบในภาพรวมว่า “Hybrid Model” โดยตัวแบบ Hybrid Model จะมีโครงสร้างดังรูปที่ 4.4



รูปที่ 4.4 โครงสร้างของ Hybrid Model

4.3.1 แต่เนื่องจากหลังจากการทดลองในข้อ 4.2 เราได้มีการเพิ่มข้อมูลใหม่ ๆ เข้าไปในชุดข้อมูลสำหรับเรียนรู้ ทำให้ชุดข้อมูลมีขนาดเพิ่มขึ้นจาก 390 ชุดเป็นประมาณ 1000 ชุด รวมถึงมีการตัดข้อมูลที่เป็น Noise ออก ซึ่งเมื่อข้อมูลมีขนาดเพิ่มขึ้นเราจึงได้ทำการทดสอบตัวแบบสำหรับคัดแยกกับชุดข้อมูลใหม่ ซึ่งได้ผลลัพธ์ดังตารางที่ 4.9

Model	Average Accuracy	Average Precision	Average Recall	Average F1 Score
Decision Tree	0.28	0.2	0.2	0.2
K Nearest Neighbors (KNN)	0.33	0.28	0.27	0.27
Random Forest	0.35	0.24	0.25	0.24
Gradient Boosting	0.34	0.2	0.23	0.21

ตารางที่ 4.9 ผลลัพธ์ของตัวแบบคัดแยกเมื่อมาทดสอบกับชุดข้อมูลใหม่*

*มีการตัด Naïve Bayes ออก เนื่องจากจากการทดลองในข้อ 4.2 จะเห็นว่า ผลลัพธ์ของ Naïve Bayes ให้ผลลัพธ์ที่ไม่น่าพอใจนัก จึงได้มีการตัดออกไปเพื่อลดเวลาการทำงานกับตัวแบบ

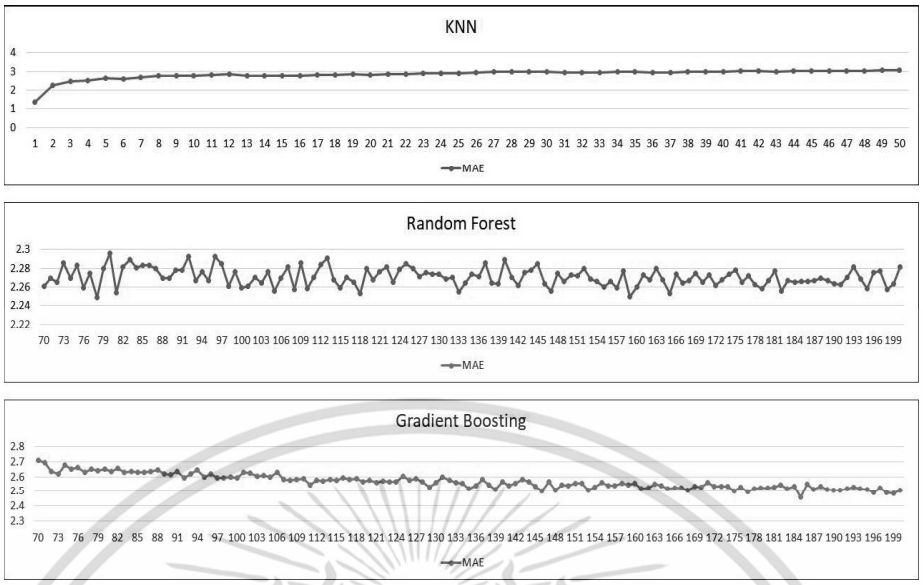
จากผลลัพธ์จะเห็นได้ว่าประสิทธิภาพของตัวแบบนั้นได้ตกลงไปอย่างเห็นได้ชัด และไม่ว่าเราจะพยายามปรับแต่งที่ตัวแบบขนาดไหนผลลัพธ์ก็ไม่ค่อยต่างไปจากเดิน เราจึงได้หันกลับไปปรับแต่งที่ชุดข้อมูลแทน โดยได้ทำการทดลองใส่ข้อมูล Day of Year (DOY) ลงไปซึ่งเกิดจากการคำนวณว่าวันที่ถ่ายรูปนั้นเป็นวันที่เท่าไรของปีนั้นเช่น 1 มกราคม จะมีค่า DOY = 1 หรือ 31 สิงหาคม จะมีค่า DOY อยู่ที่ 243 หรือ 244 ขึ้นอยู่กับว่าปีนั้นเป็นปีอธิกสุรทินหรือไม่ (เดือนกุมภาพันธ์มี 29 หรือไม่) โดยสาเหตุที่เลือกเพิ่มข้อมูล DOY ลงไปนั้นเนื่องจากการเกิดปัญหามลพิษทางอากาศของกรุงเทพมหานครและปริมณฑลนั้นมักจะเกิดขึ้นเฉพาะช่วงหน้าหนาวเป็นหลัก (พฤศจิกายน - กุมภาพันธ์) ซึ่งเราสามารถเจาะจงโดยใช้ข้อมูล DOY ได้ ซึ่งการนำข้อมูล DOY เข้าไปในชุดข้อมูลน่าจะทำให้ประสิทธิภาพในการคัดแยกนั้นเพิ่มขึ้น และอีกทั้งยังได้รับคำแนะนำว่าข้อมูลที่เป็นคุณภาพอากาศระดับ “Very Unhealthy” นั้นมีข้อมูลจำนวนน้อยมากเมื่อเทียบกับข้อมูลในระดับอื่น ๆ ซึ่งอาจจะทำให้ตัวแบบนั้นตรวจจับเป็นสิ่งรบกวนได้ จึงได้ยุบรวมข้อมูลระดับคุณภาพอากาศระดับ “Unhealthy” กับ “Very Unhealthy” รวมกัน และหลังจากนั้นจึงได้มีการทดลองนำไปใช้ในการวัดประสิทธิภาพกับตัวแบบสำหรับทำนายระดับคุณภาพอากาศ โดยโครงสร้างของตัวแบบยังอิงจากตารางที่ 4.2 และได้ผลลัพธ์ดังนี้

Model	Average Accuracy	Average Precision	Average Recall	Average F1 Score
Decision Tree	0.69	0.7	0.7	0.7
K Nearest Neighbors (KNN)	0.8	0.79	0.81	0.8
Random Forest	0.7	0.7	0.68	0.68
Gradient Boosting	0.74	0.78	0.76	0.77

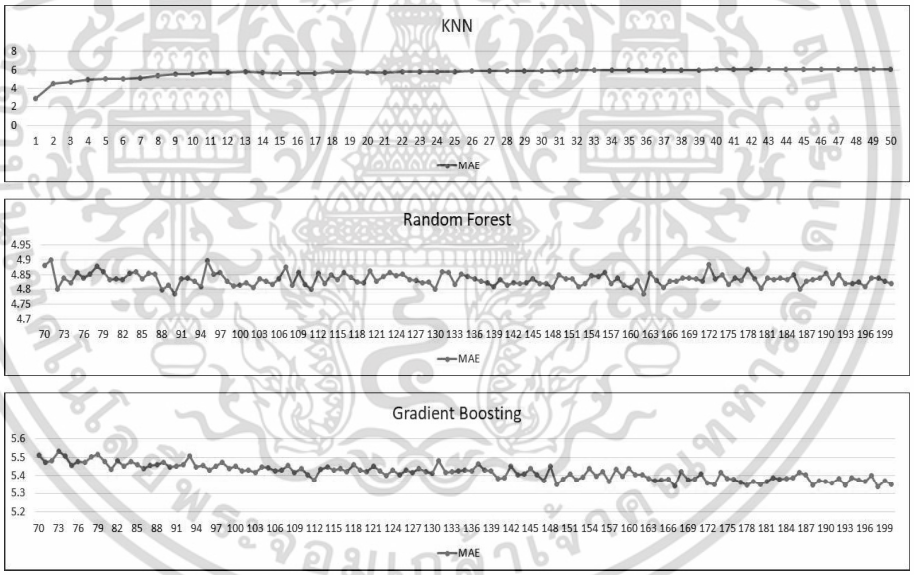
ตารางที่ 4.10 ผลลัพธ์ของตัวแบบคัดแยกเมื่อมาทดสอบกับชุดข้อมูลที่มีการเพิ่ม DOY เข้ามาแล้ว

จากตารางที่ 4.10 จะเห็นได้ว่าตัวแบบที่ดีที่สุดได้เปลี่ยนไปจาก Random Forest เป็น KNN ซึ่งใกล้เคียงกับผลลัพธ์ของข้อมูลชุดเก่า (ผลลัพธ์เดิม KNN เป็นลำดับที่ 2) จึงสรุปได้ว่าในส่วนที่เป็นตัวแบบสำหรับคัดแยกข้อมูลนั้นเราได้เลือก KNN เมื่อค่า K = 1

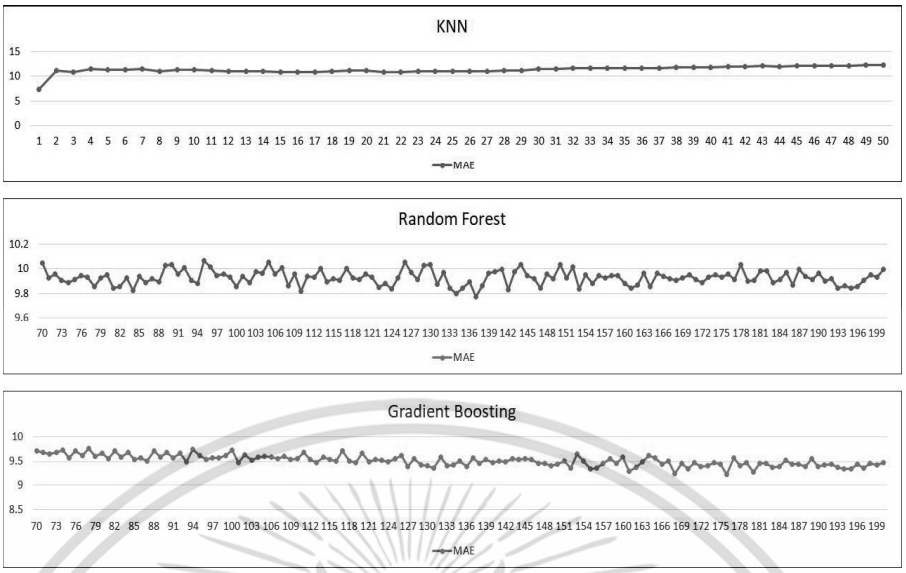
4.3.2 หลังจากได้ข้อสรุปเรื่องชุดข้อมูล และได้ตัวแบบสำหรับคัดแยกข้อมูลแล้วเราจึงเริ่มดำเนินการสร้างตัวแบบถดถอยสำหรับทำนายค่าคุณภาพอากาศ โดยจะต้องสร้างทั้งหมด 4 ตัวแบบเพื่อรองรับไปยังแต่ละระดับคุณภาพอากาศ และตัวแบบแต่ละตัวที่รองรับในแต่ละระดับคุณภาพอากาศนั้น จะถูกฝึกสอนด้วยชุดข้อมูลเฉพาะที่เป็นข้อมูลในคุณภาพอากาศนั้น ๆ เพื่อลดความผิดพลาด ซึ่งการฝึกสอนตัวแบบแต่ละตัวนั้นก็ยังจำเป็นจะต้องมีการปรับแต่งค่าพารามิเตอร์เหมือนดังเช่นตอนทำการฝึกสอนตัวแบบคัดแยก ซึ่งช่วงค่าสำหรับการปรับแต่งนั้นอ้างอิงตามตารางที่ 4.1 โดยการปรับแต่งค่า K หรือจำนวนตัวแบบถดถอยสำหรับทำนายนั้นเราจะเจาะจงไปที่ค่า Mean Absolute Error เป็นหลัก และได้ผลลัพธ์ดังรูปที่ 4.5 – 4.8 และตารางที่ 4.11 และตารางที่ 4.12



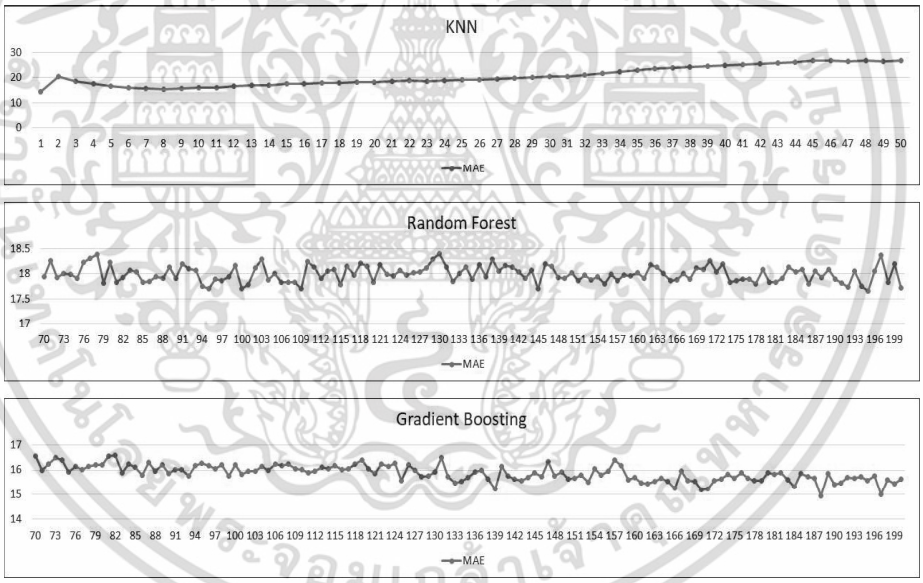
รูปที่ 4.5 ผลลัพธ์ของการปรับแต่งค่า K หรือ N Estimators สำหรับตัวแบบถดถอยที่รองรับคุณภาพอากาศในระดับ “Very Good”



รูปที่ 4.6 ผลลัพธ์ของการปรับแต่งค่า K หรือ N Estimators สำหรับตัวแบบถดถอยที่รองรับคุณภาพอากาศในระดับ “Good”



รูปที่ 4.7 ผลลัพธ์ของการปรับแต่งค่า K หรือ N Estimators สำหรับตัวแบบถดถอยที่รองรับคุณภาพอากาศในระดับ “Satisfactory”



รูปที่ 4.8 ผลลัพธ์ของการปรับแต่งค่า K หรือ N Estimators สำหรับตัวแบบถดถอยที่รองรับคุณภาพอากาศในระดับ “Unhealthy”

ระดับบคุณภาพอากาศที่ถูกทำนาย	ตัวแบบ	ค่า K หรือ N Estimators ที่ดีที่สุด
“Very Good”	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 78
	Gradient Boosting	Estimators = 185
“Good”	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 165
	Gradient Boosting	Estimators = 198
“Satisfactory”	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 195
	Gradient Boosting	Estimators = 188
“Unhealthy”	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 98
	Gradient Boosting	Estimators = 191

ตารางที่ 4.11 ผลลัพธ์ของค่าปรับแต่งที่ดีที่สุดของ KNN, Random Forest และ Gradient Boosting ของแต่ละตัวแบบที่รองรับตามระดับคุณภาพของอากาศ

หลังจากนั้นเราจึงได้ทดลองสร้างตัวแบบสำหรับแต่ละระดับคุณภาพอากาศโดยการปรับแต่งค่าพารามิเตอร์ของตัวแบบเป็นไปตามตารางที่ 4.12 และได้ผลลัพธ์ดังตารางที่ 4.13

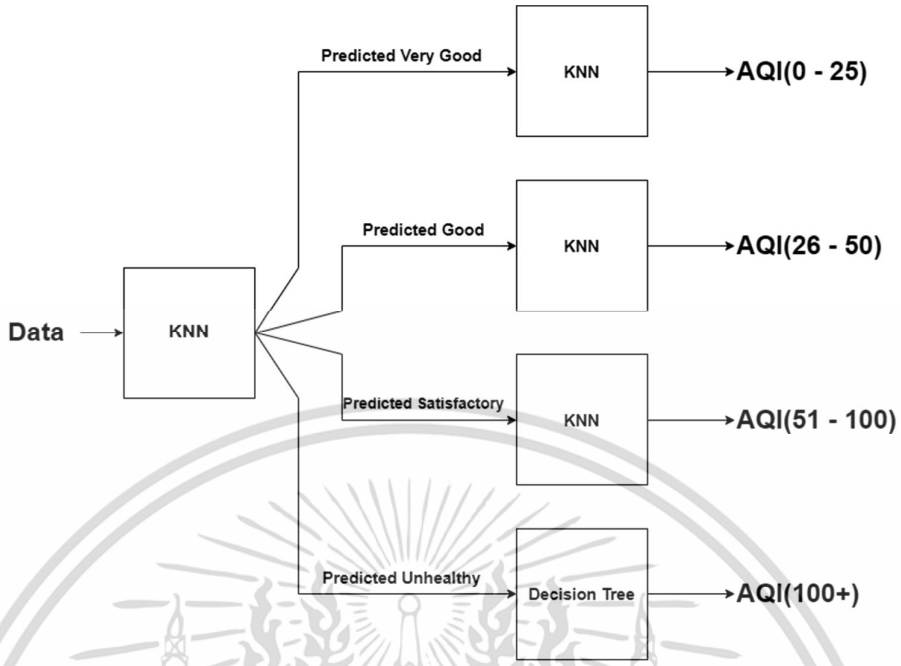
ระดับคุณภาพอากาศ	ตัวแบบ	ค่าปรับแต่ง
“Very Good”	Linear Regression	-
	Decision Tree	Criterion = MAE
	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 78, Criterion = MAE
	Gradient Boosting	Estimators = 185, Loss = MAE
“Good”	Linear Regression	-
	Decision Tree	Criterion = MAE
	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 162, Criterion = MAE
	Gradient Boosting	Estimators = 198, Loss = MAE
“Satisfactory”	Linear Regression	-
	Decision Tree	Criterion = MAE
	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 195, Criterion = MAE
	Gradient Boosting	Estimators = 188, Loss = MAE
“Unhealthy”	Linear Regression	-
	Decision Tree	Criterion = MAE
	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 98, Criterion = MAE
	Gradient Boosting	Estimators = 191, Loss = MAE

ตารางที่ 4.12 การปรับแต่งค่าพารามิเตอร์ของตัวแบบสำหรับทำนายค่าคุณภาพอากาศที่รองรับในแต่ละระดับคุณภาพอากาศ

ระดับคุณภาพอากาศ	ตัวแบบ	MAE	R^2
“Very Good”	Linear Regression	3.724812827	-0.007804389
	Decision Tree	1.696	0.479183808
	K-Nearest Neighbors	1.36	0.586967254
	Random Forest	2.211179487	0.569412096
	Gradient Boosting	2.33029211	0.454437082
“Good”	Linear Regression	5.870447831	0.011584563
	Decision Tree	4.674418605	0.09720783
	K-Nearest Neighbors	2.918604651	0.363667765
	Random Forest	4.722437554	0.305527453
	Gradient Boosting	5.174536175	0.123705025
“Satisfactory”	Linear Regression	12.01142116	-0.033751627
	Decision Tree	11.19565217	-0.477303958
	K-Nearest Neighbors	7.326086957	0.023221455
	Random Forest	9.558235481	0.308781089
	Gradient Boosting	8.713142288	0.412421994
“Unhealthy”	Linear Regression	28.30169912	-0.239189192
	Decision Tree	12.55263158	0.392962918
	K-Nearest Neighbors	15.98245614	0.445747726
	Random Forest	17.24520918	0.454456633
	Gradient Boosting	13.87447744	0.592495801

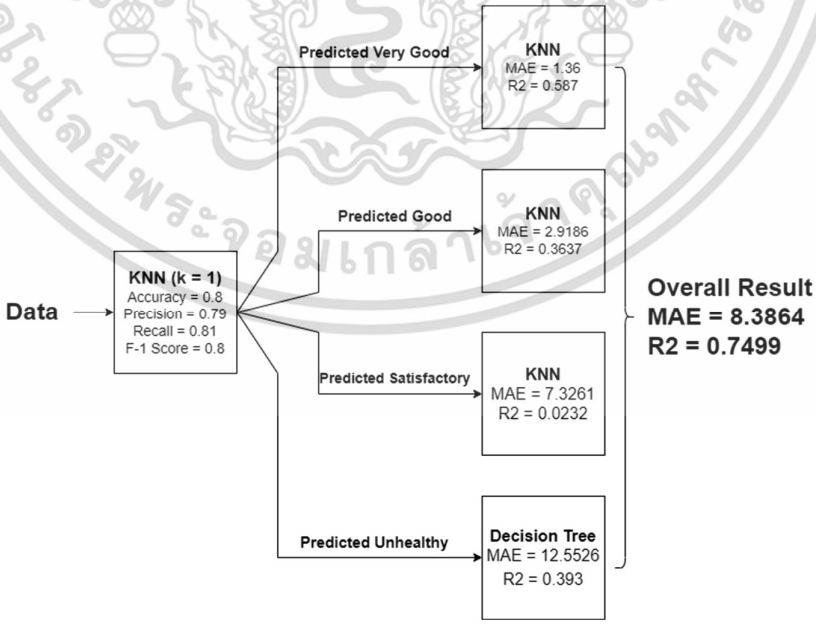
ตารางที่ 4.13 ผลลัพธ์ประสิทธิภาพของตัวแบบสำหรับทำนายค่าคุณภาพอากาศที่รองรับในแต่ละระดับคุณภาพอากาศ

จากตารางที่ 4.13 เราจึงทำการเลือกตัวแบบถดถอยสำหรับแต่ละระดับคุณภาพอากาศโดยใช้เกณฑ์ค่า MAE เป็นตัวตัดสิน และได้ผลสรุปดังนี้ 1. ระดับ “Very Good” เลือก KNN ด้วยผลลัพธ์ MAE = 1.36, 2. ระดับ “Good” เลือก KNN ด้วยผลลัพธ์ MAE = 2.9186, 3. ระดับ “Satisfactory” เลือก KNN ด้วยผลลัพธ์ MAE = 7.3261 และ 4. ระดับ “Unhealthy” เลือก Decision Tree ด้วยผลลัพธ์ MAE = 12.5526 และเมื่อรวมกับผลลัพธ์จากข้อ 4.3.1 เพื่อประกอบตัวแบบให้ได้ตามตัวแบบที่เราคาดหวังไว้ในรูปที่ 4.4 จะได้ผลลัพธ์ดังรูปที่ 4.9



รูปที่ 4.9 ตัวแบบสำหรับทำนายค่าคุณภาพอากาศตามที่เราคาดหวัง

จากนั้นเราจึงได้ทำการวัดประสิทธิภาพของตัวแบบโดยรวมโดยนำข้อมูลที่แบ่งไว้จากชุดฝึกสอนในตอนแรก (ที่แบ่งไว้ 30%) มาทำการทดสอบและได้ผลลัพธ์ ค่า Mean Absolute Error (MAE) = 8.3864 และค่า Coefficient of Determination (R^2) = 0.7499 ซึ่งเป็นผลลัพธ์ที่น่าพอใจอย่างมาก ดังรูปที่ 4.10

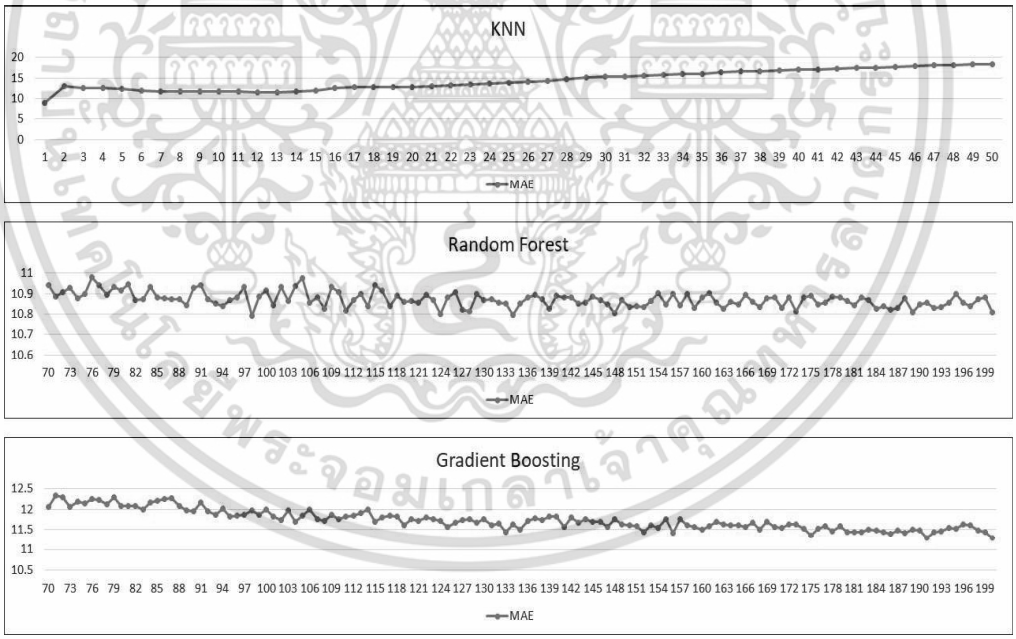


รูปที่ 4.10 ผลลัพธ์โดยรวมของตัวแบบสำหรับทำนายตามที่เราคาดหวังไว้

นอกจากนั้นเรายังได้มีการคำนวณในเฉพาะส่วนการทำนายค่าคุณภาพอากาศ กล่าวคือเราสมมติว่าถ้าผลลัพธ์จากการคัดแยกข้อมูลนั้นถูกต้องทั้งหมดแล้ว ในส่วนที่ทำนายออกมาเป็นค่าคุณภาพอากาศจะได้ผลลัพธ์เป็น Mean Absolute Error (MAE) = 4.3567 และ Coefficient of Determination (R^2) = 0.9468 ซึ่งถือว่าเป็นผลลัพธ์ที่ยอดเยี่ยมมาก

4.4 จากนั้นด้วยความสงสัย และต้องการจะทำการทดลองเพื่อเปรียบเทียบ จึงได้ทำการสร้างตัวแบบสำหรับทำนาย โดยที่ใช้แต่ตัวแบบถดถอยเท่านั้นโดยเราจะเรียกตัวแบบนี้ว่า “Pure Regressor Model” และการฝึกสอนเราจะใช้ข้อมูลชุดเดียวกันกับที่ทดลองในข้อ 4.3 มาทำการฝึกสอนและทำการทดสอบ เพื่อจะลดความคลาดเคลื่อนในการเปรียบเทียบประสิทธิภาพของทั้งสองตัวแบบ

การปรับแต่งพารามิเตอร์ของตัวแบบถดถอยนั้นสำหรับ KNN, Random Forest และ Gradient Boosting นั้นจะต้องมีการทำการหาค่า K หรือ N Estimators ที่เหมาะสม เหมือนดังตัวแบบในข้อ 4.3 โดยการหาค่า K หรือ N Estimators ที่เหมาะสมนั้นจะใช้ช่วงการหาจะอิงตามตารางที่ 4.1 โดยการหาค่า K หรือ N Estimators ที่เหมาะสมนั้นได้ผลลัพธ์ดังรูปที่ 4.11 และตารางที่ 4.14



รูปที่ 4.11 ผลลัพธ์ของการปรับแต่งค่า K หรือ N Estimators สำหรับตัวแบบ Pure Regression Model

ระดับคุณภาพอากาศ	ตัวแบบ	ค่าปรับแต่ง
Pure Regressor model	K-Nearest Neighbors	k = 1
	Random Forest	Estimators = 98
	Gradient Boosting	Estimators = 191

ตารางที่ 4.14 ผลลัพธ์ของค่าปรับแต่งที่ดีที่สุดของ KNN, Random Forest และ Gradient Boosting ของ Pure Regression Model

และได้วิธีการปรับค่าพารามิเตอร์ของตัวแบบดังตารางที่ 4.15 และได้ผลลัพธ์ประสิทธิภาพของแต่ละตัวแบบถดถอยสำหรับทำนายค่าคุณภาพอากาศดังตารางที่ 4.16

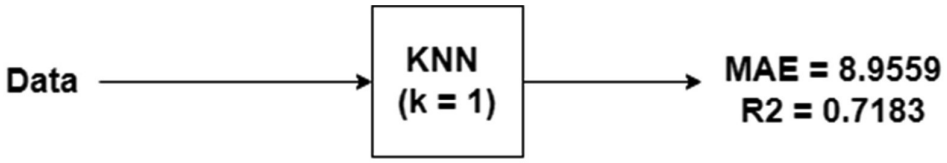
ตัวแบบ	ค่าปรับแต่ง
Linear Regression	-
Decision Tree	Criterion = MAE
K-Nearest Neighbors	k = 1
Random Forest	Estimators = 98, Criterion = MAE
Gradient Boosting	Estimators = 191, Loss = MAE

ตารางที่ 4.15 การปรับแต่งค่าพารามิเตอร์ของตัวแบบสำหรับทำนายค่าคุณภาพอากาศของ Pure Regression Model

ตัวแบบ	MAE	R^2
Linear Regression	28.1998383	0.20500946
Decision Tree	12.40677966	0.634164936
K-Nearest Neighbors	8.955932203	0.718283228
Random Forest	10.59320304	0.832397213
Gradient Boosting	10.808192	0.813546129

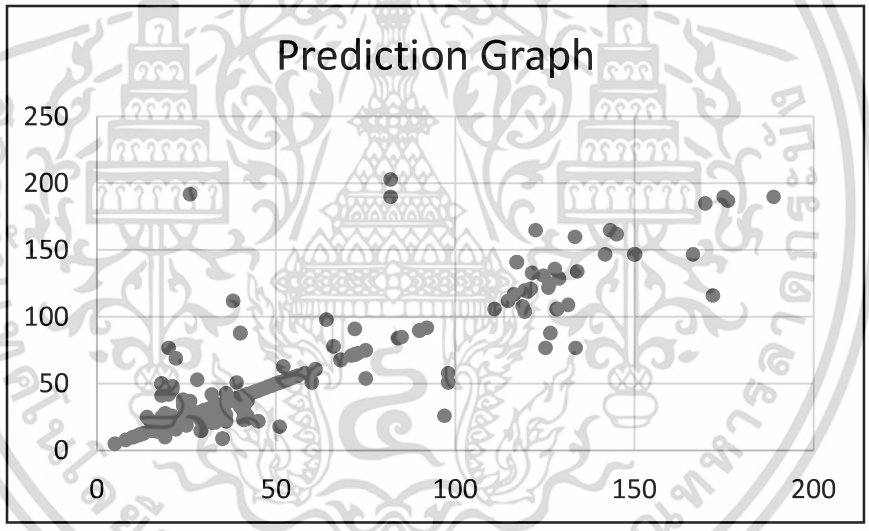
ตารางที่ 4.16 ผลลัพธ์ประสิทธิภาพของตัวแบบสำหรับทำนายค่าคุณภาพอากาศของ Pure Regression Model

จากตารางที่ 4.16 จะเห็นได้ว่าหากเราใช้ MAE เป็นเกณฑ์ในการวัดประสิทธิภาพจะเห็นได้ว่า K-Nearest Neighbors หรือ KNN เมื่อค่า K = 1 นั้นให้ประสิทธิภาพที่ดีที่สุดที่ MAE = 8.9559 และ R^2 = 0.7183 ทำให้ได้ตัวแบบสำหรับ Pure Regression Model ดังรูปที่ 4.12

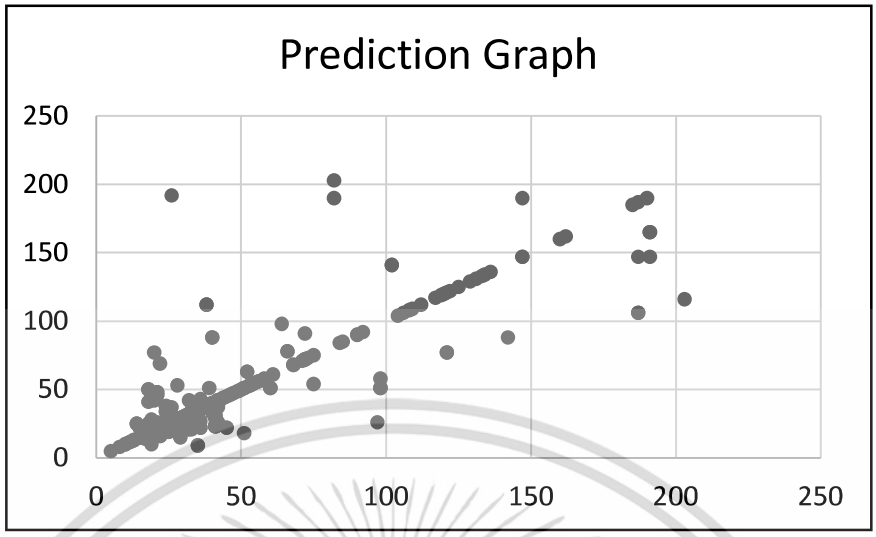


รูปที่ 4.12 ผลลัพธ์โดยรวมของตัวแบบสำหรับทำนายที่ใช้เฉพาะตัวแบบถดถอย และชุดข้อมูลที่ปรับปรุงล่าสุด

4.5 จากการเปรียบเทียบผลลัพธ์ในข้อ 4.3 (รูปที่4.10) และข้อ 4.4 (รูปที่ 4.12) จะเห็นได้ว่าทั้งสองตัวแบบนั้นมีประสิทธิภาพที่ใกล้เคียงกันมาก เราจึงได้ทำการพล็อตกราฟการทำนาย (Predicted Graph) ของแต่ละตัวแบบ เพื่อสังเหตุ และวิเคราะห์ว่ากราฟใดจะมีลักษณะของข้อมูลอยู่ในลักษณะของเส้นตรงมากกว่ากัน โดยหากตัวแบบให้ผลลัพธ์การทำนายที่มีประสิทธิภาพมาก กราฟจะรวมกลุ่มกันอยู่ในลักษณะเส้นตรงทแยงมุม และได้ผลลัพธ์กราฟการทำนายดังรูปที่ 4.13 และ 4.14



รูปที่ 4.13 กราฟการทำนายของตัวแบบที่เรานำเสนอในวิทยานิพนธ์นี้



รูปที่ 4.14 กราฟการทำนายของตัวแบบที่ใช้เฉพาะตัวแบบถดถอย

จากกราฟในรูปที่ 4.13 และ 4.14 นั้นจะเห็นได้ว่ากราฟของทั้งสองตัวแบบมีลักษณะคล้ายคลึงกันมาก และกราฟของทั้งสองตัวแบบ ก็มีลักษณะที่มีการกระจายตัวในลักษณะเส้นตรงทแยงมุม ซึ่งก็ถือว่ามีประสิทธิภาพดีมากทั้งสองตัวแบบ

บทที่ 5

สรุปผลการทดลอง และอภิปรายผล

ในวิทยานิพนธ์นี้ เราได้นำเสนอวิธีการประเมินค่าคุณภาพด้วยวิธีใหม่ โดยการใช้ข้อมูลจากภาพถ่ายดาวเทียมร่วมกับการเรียนรู้ของเครื่อง โดยในขั้นแรกเราได้ทดลองนำเฉพาะข้อมูลภาพถ่ายดาวเทียม และข้อมูล Vegetable Indices ซึ่งคำนวณได้จากข้อมูลภาพถ่ายดาวเทียม มาทำการทดลองกับการเรียนรู้ของเครื่องในลักษณะของตัวแบบถดถอย แต่ไม่ประสบผลลัพธ์ที่น่าพอใจ

ต่อมาเราจึงได้ทำการนำข้อมูลดังกล่าวมาทำการทดลองกับการเรียนรู้ของเครื่องในลักษณะตัวแบบคัดแยกสำหรับคัดแยกข้อมูลออกเป็นแต่ละระดับคุณภาพอากาศซึ่งตามมาตรฐานคุณภาพอากาศของกรมควบคุมมลพิษประเทศไทยนั้นจะมีคุณภาพอากาศอยู่ 5 ระดับดังตารางที่ 2.3 โดยได้ผลลัพธ์ว่าตัวแบบ Random Forest ให้ผลลัพธ์ที่ดีที่สุดดังตารางที่ 4.8 ซึ่งเราได้นำเสนอผลลัพธ์นี้ที่งานประชุมวิชาการ ECTI-CON 2020 หลังจากเราเห็นว่าเราสามารถคัดแยกข้อมูลไปยังคุณภาพอากาศต่าง ๆ ได้ เราจึงได้ออกแบบต่อมาเพื่อให้ตัวแบบโดยรวมนั้นสามารถทำนายค่าคุณภาพอากาศได้ ซึ่งได้แบบร่างตัวแบบโดยรวมดังรูปที่ 4.4 ซึ่งตัวแบบที่ร่างออกมานั้นเราเรียกว่า “Hybrid Model”

แต่ก่อนหน้าที่เราจะดำเนินการขั้นต่อไป เนื่องจากเราได้มีการเพิ่มข้อมูลเข้าไปยังชุดข้อมูลสำหรับฝึกสอน ทำให้มีข้อมูลจำนวนมากกว่าเดิมประมาณ 3 เท่า รวมถึงมีการตัดข้อมูลที่เป็น Noise ออกไปด้วย ทำให้เราต้องทำการทดสอบกับตัวแบบในส่วนคัดแยกอีกครั้งเพื่อตรวจสอบว่าผลลัพธ์ยังเป็นเช่นเดิมหรือไม่ ซึ่งผลลัพธ์ที่ออกมานั้นเป็นผลที่ไม่พอใจอีกครั้งดังตารางที่ 4.9 ในครั้งนี้เราจึงได้ทำการเพิ่มข้อมูลวันที่ถ่ายภาพ โดยคำนวณให้เป็นวันที่ในปีนั้น ๆ หรือที่เรียกว่า Day of Year อีกทั้งยังต้องยุบรวมข้อมูลคุณภาพอากาศในระดับ “Very Unhealthy” มารวมกับ “Unhealthy” เนื่องจากข้อมูลในระดับ Very Unhealthy มีจำนวนน้อยมาก ซึ่งอาจจะทำให้ตัวแบบคัดแยกมองว่าเป็นข้อมูลรบกวนได้ (Noise) หลังจากนั้นเราจึงนำข้อมูลชุดใหม่เข้าทำการทดลองคัดแยกและได้ผลลัพธ์ว่าตัวแบบ KNN เมื่อ K = 1 ให้ผลลัพธ์ดีที่สุดที่ Average Accuracy = 0.8, Average Precision = 0.79, Average Recall = 0.81 และ Average F1 Score = 0.8 ดังผลลัพธ์ที่แสดงในตารางที่ 4.10

หลังจากนั้นเราจึงได้นำข้อมูลชุดล่าสุดมาสร้างตัวแบบถดถอยสำหรับการทำนายค่าคุณภาพอากาศจากข้อมูลที่ได้รับการคัดแยกมาก่อนหน้า ซึ่งตามตัวแบบที่เราคาดหวัง (ภาพ 4.4) จะเห็นได้ว่าเราต้องสร้างตัวแบบถดถอยทั้งหมด 4 ตัว เพื่อรองรับกับข้อมูลที่จะถูกแยกออกมา 4 ระดับ โดยตัววัด

ประสิทธิภาพที่เราใช้เป็นหลักคือค่า Mean Absolute Error (MAE) ซึ่งจากผลลัพธ์การเรียนรู้เราสรุปผลได้ดังนี้ 1. ระดับ “Very Good” ตัวแบบที่เลือกได้แก่ตัวแบบ KNN เมื่อ K = 1 ด้วยผลลัพธ์ MAE = 1.36, 2. ระดับ “Good” ตัวแบบที่เลือกได้แก่ตัวแบบ KNN เมื่อ K = 1 ด้วยผลลัพธ์ MAE = 2.9186, 3. ระดับ “Satisfactory” ตัวแบบที่เลือกได้แก่ตัวแบบ KNN เมื่อ K = 1 ด้วยผลลัพธ์ MAE = 7.3261, 4. ระดับ “Unhealthy” ตัวแบบที่เลือกได้แก่ตัวแบบ Decision Tree ด้วยผลลัพธ์ MAE = 12.5526 และเมื่อเรานำส่วนของการคัดแยกประกอบเข้ากับส่วนทำนายค่าคุณภาพแล้ว และทำการทำนายผลลัพธ์ทั้งหมดได้ผลลัพธ์ดังนี้ ตัวแบบ Hybrid Model MAE = 8.3864 และ R^2 = 0.7499 ดังผลลัพธ์ในรูปที่ 4.10 และมีผลลัพธ์เฉพาะส่วนทำนายค่าคุณภาพอากาศ (โดยนับว่าข้อมูลผ่านตัวคัดแยกมาถูกต้องทั้งหมด) MAE = 4.3567 และ R^2 = 0.9468

ต่อมาเราจึงได้ทดลองสร้างตัวแบบสำหรับทำนายค่าคุณภาพอากาศที่ใช้เฉพาะตัวแบบถดถอย ซึ่งในงานวิจัยนี้เราเรียกว่า Pure Regression Model จากนั้นจึงทดลองฝึกสอนตัวแบบอีกครั้งด้วยข้อมูลที่เราปรับปรุงล่าสุด เพื่อนำผลลัพธ์มาเปรียบเทียบกับผลลัพธ์ของตัวแบบ Hybrid Model จากการฝึกสอนเราผลลัพธ์ว่า ตัวแบบ KNN เมื่อ K = 1 ให้ผลลัพธ์ดีที่สุดด้วยผลลัพธ์ MAE = 8.9559 และ R^2 = 0.7183 ดังผลลัพธ์ในตารางที่ 4.16 และรูปที่ 4.12

จากผลลัพธ์ของทั้ง Hybrid Model และ Pure Regression Model สังเกตุได้ว่ามีผลลัพธ์ที่ใกล้เคียงกันมาก ซึ่งยังทำให้เราไม่สามารถสรุปได้ทันทีว่าตัวแบบไหนให้ผลลัพธ์ได้ดีกว่ากัน เราจึงได้ทำการสร้างกราฟผลการทำนาย (Prediction Graph) เพื่อสังเกตการณ์กระจายตัวของข้อมูลว่ากระจายตัวอยู่ในลักษณะของเส้นตรงทแยงมุมหรือไม่ โดยกราฟผลการทำนายแสดงอยู่ในรูปที่ 4.13 และ 4.14 ซึ่งจากลักษณะการประจายตัวของทั้งสองกราฟจะเห็นได้ว่า มีการกระจายตัวเกาะกลุ่มในลักษณะเส้นตรงกันทั้งคู่

จากผลลัพธ์ที่ได้กล่าวมาทั้งหมดทางเราจึงสรุปได้ว่าในงานนี้ Hybrid Model ที่เกิดจากการผสมผสานวิธีการคัดแยกและการทำนายค่าของข้อมูล มีประสิทธิภาพมากกว่า Pure Regression Model ที่เกิดจากวิธีการทำนายค่าข้อมูลเพียงอย่างเดียว

ในที่นี้เราจะมากล่าวถึงจุดที่สามารถพัฒนาต่อยอด ซึ่งจากงานนี้เราสามารถต่อยอดออกไปได้หลากหลายเส้นทางมาก เช่น หากนำวิธีการแบบ Hybrid ของเราไปใช้กับชุดข้อมูลเชิงพื้นที่แบบในงานวิจัยที่เราได้ทำการรีวิวมานั้นจะให้ผลลัพธ์ที่ดีกว่า ตัวแบบที่เกิดจากวิธีการเดียวหรือไม่ หรือไม่ว่าจะเป็นการเปลี่ยนข้อมูลดาวเทียมที่มีชุดข้อมูลเหมือนของเรา แต่มีคาบการถ่ายภาพที่ถี่กว่าของเรา

เพราะ Landsat 8 มีคาบการถ่ายที่ 16 วัน ซึ่งถือว่ายาวนานมาก แต่ก็แลกมากับความละเอียดของภาพถ่ายดาวเทียมที่สูง (Spatial Resolution 30 เมตร) ตัวอย่างดาวเทียมที่มีคาบการถ่ายเร็วกว่า Landsat 8 เช่น Himawari-8 มีคาบการถ่ายที่ 15 นาที แต่มี Spatial Resolution ที่ 500 เมตร เป็นต้น

อีกข้อหนึ่งที่สามารถพัฒนาต่อยอดได้คือ ณ เวลาปัจจุบันมีการเพิ่มสถานีตรวจวัดแล้ว ทำให้เรามีข้อมูลจริงมากขึ้น และจะทำให้ชุดข้อมูลมีมากขึ้นด้วย เมื่อข้อมูลมีจำนวนมากเราก็สามารถไปทดลองกับการเรียนรู้เชิงลึกได้ (Deep Learning) ที่งานวิจัยนี้ไม่ทดลองกับการเรียนรู้เชิงลึกเนื่องจากได้มีการปรึกษากับผู้เชี่ยวชาญที่ทำงานวิจัยเกี่ยวกับการเรียนรู้เชิงลึกแล้ว และได้รับคำแนะนำว่าข้อมูลยังมีจำนวนน้อยเกินไปไม่เหมาะใช้กับการเรียนรู้เชิงลึก โดยเกรงว่าอาจจะเกิดการ Overfit ขึ้นได้

อีกข้อหนึ่งคือจากผลลัพธ์ของ Hybrid Model ในส่วยของการทำนายค่าคุณภาพอากาศเพียงอย่างเดียวนั้นจะเห็นได้ว่ามีประสิทธิภาพสูงมาก (MAE = 4.3567 และ R^2 = 0.9468) ซึ่งการที่ประสิทธิภาพมีการตกลงไปนั้นเนื่องจากมีการทำนายที่ผิดพลาดในส่วนคัดแยก ซึ่งจุดนี้หากมีการพัฒนาต่อยอดในส่วนคัดแยกให้มีประสิทธิภาพมากขึ้น ตัวแบบเองก็น่าจะมีผลลัพธ์ที่ดีขึ้นด้วยเช่นกัน

บรรณานุกรม

1. https://en.wikipedia.org/wiki/PostgreSQL

2. https://www.opendatacube.org/

3. https://landsat.gsfc.nasa.gov/landsat-data-continuity-mission/

4. Rouse et al. (1974). Monitoring vegetation systems in the Great Plains with ERTS. In: S.C. Freden, E.P. Mercanti, and M. Becker (eds) Third Earth Resources Technology Satellite–1 Syposium. Volume I: Technical Presentations, 1974, 309-317.

5. McDaniel K C and Haas R H (1982). Assessing mesquite-grass vegetation condition from Landsat. Photo Eng Remote Sens. 48. 441–450.

6. กรมควบคุมมลพิษ กระทรวงทรัพยากรณ์ธรรมชาติและสิ่งแวดล้อม ประเทศไทย

7. Leo Breiman (2001). Random Forest. Machine Learning. 45, 5-32.

8. Jerome H. Friendman (2001). GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE. The Annals of Statistics. 1189 – 1232.

9. Manliguez, C. (2016) Generalized Confusion Matrix for Multiple Classes. [online] จากเว็บไซต์< https://www.researchgate.net/publication/310799885_Generalized_Confusion_Matrix_for_Multiple_Classes> [เข้าถึงเมื่อวันที่ 30 พฤษภาคม 2022].

10. https://en.wikipedia.org/wiki/Coefficient_of_determination

11. Chitrini Mozumder et al. (2012). Air Pollution Modeling from Remotely Sensed Data Using Regression Techniques. Indian Society of Remote Sensing. 269 – 277. https://doi.org/10.1007/s12524-012-0235-2.

12. Qian Di et al. (2016). Assessing PM2.5 Exposures with High Spatiotemporal Resolution across the Continental United States. Environ Sci Technol. 4712 – 4721. https://doi.org/10.1021/acs.est.5b06121.

13. Husanbir Singh Pannu et al. (2017). Multi-objective particle swarm optimization-based adaptive neuro-fuzzy inference system for benzene monitoring. The Natural Computing Applications Forum. 2195 – 2205. https://doi.org/10.1007/s00521-017-3181-7

14. Mehdi Zamani Joharestani et al. (2019). PM2.5 Prediction Based on Random Forest, XGBoost, and Deep Learning Using Multisource Remote Sensing Data. Atmosphere. https://doi.org/10.3390/atmos10070373

15. Jasleen Kuar Sethi and Mamta Mittal (2019). Ambient Air Quality Estimation using Supervised Learning Techniques. EAI Endorsed Transactions on Scalable Information Systems. 1 – 10. https://doi.org/10.4108/eai.13-7-2018.159406

16. Weilin Wang et al. (2019). Estimation of PM2.5 Concentrations in China Using a Spatial Back Propagation Neural Network. Scientific Reports. https://doi.org/10.1038/s41598-019-50177-1

17. Xiankun Sun et al. (2020). Dynamic Monitoring of Haze Pollution Using Satellite Remote Sensing. IEEE Sensors Journal. 11802 - 11810. https://doi.org/10.1109/JSEN.2019.2963158

18. W. C. Leong et al. (2020). Prediction of air pollution index (API) using support vector machine (SVM). Journal of Environmental Chemical Engineering. https://doi.org/10.1016/j.jece.2019.103208

19. Yuanlin Gu et al. (2022). Hybrid interpretable predictive machine learning model for air pollution prediction. Neurocomputing. 468. 123-136. https://doi.org/10.1016/j.neucom.2021.09.051

20. Fangzhou Lin et al. (2022). An Effective Convolutional Neural Network for Visualized Understanding Transboundary Air Pollution Based on Himawari-8 Satellite Images. IEEE Geoscience and Remote Sensing Letters. https://doi.org/10.1109/LGRS.2021.3096629

21. Huimin Ji et al. (2022). Research on adaption to air pollution in Chinese cities: Evidence from social media-based health sensing. Environment Research. https://doi.org/10.1016/j.envres.2022.112762

22. Marc Saez and Maria A. Barcelo (2022). Spatial prediction of air pollution levels using a hierarchical Bayesian
spatiotemporal model in Catalonia, Spain. Environmental Modelling and Software. https://doi.org/10.1016/j.envsoft.2022.105369

23. Andrews A. Boateng (2022). Evaluation of chemometric classification and regression models for the detection of syrup adulteration in honey. LWT - Food Science and Technology. https://doi.org/10.1016/j.lwt.2022.113498
24. http://air4thai.pcd.go.th/webV3/#/Report