



## โครงการ

# การเรียนรู้การสอนเพื่อเสริมประสบการณ์

ชื่อโครงการ การประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อ  
โควิด-19 ในประเทศไทย

Graph signal processing to identify the epidemic pattern  
of COVID-19 in Thailand

ชื่อนิสิต นาย ศรัณย์พัทธ ศุจินันท์ชยะกูร 6133758323

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์  
สาขาวิชา วิทยาการคอมพิวเตอร์

ปีการศึกษา 2564

คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

การประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย

นาย ศรัณย์พัทธ์ ศุจินันท์ชยะกูร

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต  
สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์  
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย  
ปีการศึกษา 2564  
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Graph signal processing to identify the epidemic pattern of COVID-19 in Thailand

Sarunpart Suginunchayagun

A Project Submitted in Partial Fulfillment of the Requirements  
for the Degree of Bachelor of Science Program in Computer Science

Department of Mathematics and Computer Science

Faculty of Science

Chulalongkorn University


Academic Year 2021

Copyright of Chulalongkorn University


หัวข้อโครงการ	การประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย
โดย	นาย ศรัณย์พัทธ์ ศุจินันท์ชยะกูร
สาขาวิชา	วิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษาโครงการหลัก	ผู้ช่วยศาสตราจารย์ ดร. อาธร เหลืองสดใส
อาจารย์ที่ปรึกษาโครงการร่วม	อาจารย์ ดร. ทรรปณ์ ปณิธานะรักษ์


---

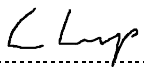
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย  
อนุมัติให้แนบโครงการฉบับนี้เป็นส่วนหนึ่ง ของการศึกษาตามหลักสูตรปริญญาบัณฑิต ในรายวิชา  
2301499 โครงการวิทยาศาสตร์ (Senior Project)

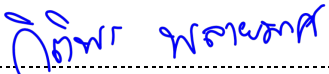
  
..... หัวหน้าภาควิชาคณิตศาสตร์  
(รองศาสตราจารย์ ดร. รตินันท์ บุญเคลือบ) และวิทยาการคอมพิวเตอร์

คณะกรรมการสอบโครงการ

  
..... อาจารย์ที่ปรึกษาโครงการหลัก  
(ผู้ช่วยศาสตราจารย์ ดร. อาธร เหลืองสดใส)

  
..... อาจารย์ที่ปรึกษาโครงการร่วม  
(อาจารย์ ดร. ทรรปณ์ ปณิธานะรักษ์)

  
..... กรรมการ  
(ศาสตราจารย์ ดร. ชิตชนก เหลือสินทรัพย์)

  
..... กรรมการ  
(ผู้ช่วยศาสตราจารย์ ดร. กิติพร พลายมาศ)

นาย ศรัณย์พัท ศุจินันท์ชยะกูร: การประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย. (Graph signal processing to identify the epidemic pattern of COVID-19 in Thailand) อ.ที่ปรึกษาโครงการหลัก : ผู้ช่วยศาสตราจารย์ ดร. อาธร เหลืองสดใส, อาจารย์ที่ปรึกษาโครงการร่วม : อาจารย์ ดร. ทรรพณ์ ปณิธานะรักษ์, 71 หน้า.

งานวิจัยนี้มีจุดประสงค์เพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย โดยอ้างอิงวิธีการมาจากงานวิจัยของ Yang Li และ Gonzalo Mateos จากการสร้างกราฟจากข้อมูลระยะห่างระหว่างอำเภอรวมทั้งสิ้น 928 อำเภอ และสร้างสัญญาณจากข้อมูลจำนวนผู้ติดเชื้อรายสัปดาห์ในแต่ละอำเภอตลอด 39 สัปดาห์นับตั้งแต่เริ่มการระบาดระลอกที่ 3 จนถึงการระบาดสงบลงหรือตั้งแต่ 1 เมษายน 2564 จนถึง 30 ธันวาคม 2564 ในการศึกษาเสนอวิธีการที่แตกต่างไปจากงานวิจัยของ Yang Li และ Gonzalo Mateos โดยสร้างกราฟสำหรับนำไปทำการประมวลผลสัญญาณกราฟด้วยข้อมูลระยะห่างระหว่างอำเภอ แทนการใช้ข้อมูล สถานที่ทำงานของประชากรในแต่ละอำเภอ (commute flow) และได้เสนอวิธีการระบุรูปแบบการระบาดด้วยกระบวนการปรับสัญญาณกราฟให้เรียบแบบในอุดมคติโดยใช้สัญญาณในช่วงความถี่ต่ำเพียงอย่างเดียว แทนการใช้ทั้งสัญญาณในช่วงความถี่ต่ำและสัญญาณในช่วงความถี่สูง ผลการศึกษาพบว่าวิธีการที่เสนอสามารถระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 โดยผลลัพธ์ที่ได้จะมีค่าชุดเดียวที่ใช้อธิบายถึงรูปแบบการระบาดทำให้การระบุรูปแบบการระบาดในแต่ละอำเภอทำได้ชัดเจนมากขึ้น และสามารถแบ่งรูปแบบการระบาดได้ละเอียดมากขึ้น โดยสามารถแบ่งรูปแบบการระบาดออกเป็น 3 ประเภทได้แก่ บริเวณที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับรอบข้าง บริเวณที่มีจำนวนผู้ติดเชื่อน้อยกว่ารอบข้าง และบริเวณที่มีจำนวนผู้ติดเชื้อมากกว่ารอบข้าง

ภาควิชา...คณิตศาสตร์และวิทยาการคอมพิวเตอร์...ลายมือชื่อนิสิต... ศรัณย์พัท ศุจินันท์ชยะกูร  
สาขาวิชา...วิทยาการคอมพิวเตอร์...ลายมือชื่อ อ.ที่ปรึกษาโครงการหลัก... อาธร เหลืองสดใส  
ปีการศึกษา... 2564...ลายมือชื่อ อ.ที่ปรึกษาโครงการร่วม... ทรรพณ์ ปณิธานะรักษ์

6133758323: MAJOR COMPUTER SCIENCE

KEYWORDS : Graph signal processing, COVID-19

SARUNPART SUGINUNCHAYAGUN: GRAPH SIGNAL PROCESSING TO IDENTIFY THE  
EPIDEMIC PATTERN OF COVID-19 IN THAILAND. ADVISOR : ASST. PROF. ARTHORN  
LUANGSODSAI, Ph.D., CO-ADVISOR : THAP PANITANARAK, Ph.D., 71 pp.

The purpose of this research was to identify the epidemic pattern of COVID-19 in Thailand, based on a methodology derived from the study of Yang Li and Gonzalo Mateos. In this method, a graph was constructed from the distance data of 928 districts and signal was constructed from data on the number of weekly cases in each district for 39 weeks from the start of the third wave of outbreaks to the end of the outbreak or from 1 April 2021 until 30 December 2021. This study proposed a different approach from Yang Li's. First, a graph was constructed for the graph signal processing from distance between district data instead of using commute flow data and proposed a method for identifying the epidemic pattern by an ideal graph signal smoothing process and using only low-frequency signals instead of using signals from both low frequency and high frequency ranges. The results of the study showed that the proposed method can identify the epidemic pattern of COVID-19. The results was a single set of values that described the epidemic pattern which made more efficiently and divided the epidemic pattern more clearly. The epidemic pattern can be divided into three types: regions with a similar infection to the neighbors, regions with less infection than the neighbors, and regions with higher infection than the neighbors.

Department : Mathematics and Computer Science Student's Signature 

Field of Study : Computer Science Advisor's Signature 

Academic Year : 2021 Co-advisor's Signature 

## กิตติกรรมประกาศ

โครงการ“การประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย” ได้รับการสนับสนุนอย่างเต็มที่จาก ผู้ช่วยศาสตราจารย์ ดร. อาธร เหลืองสดใส อาจารย์ที่ปรึกษาหลัก และอาจารย์ ดร. ทรรพณ์ ปณิธานะรักษ์ อาจารย์ที่ปรึกษาโครงการร่วม ที่คอยให้คำแนะนำ คำปรึกษา ข้อเสนอแนะ ชี้แนะแนวทางการแก้ไขปัญหา ตลอดจนปรับปรุงแก้ไข ข้อผิดพลาดและข้อบกพร่องต่าง ๆ ทำให้โครงการมีความถูกต้องสมบูรณ์มากยิ่งขึ้น ด้วยความเอาใจใส่ ตั้งแต่เริ่มดำเนินโครงการจนกระทั่งโครงการสำเร็จไปด้วยดี

ขอขอบพระคุณ ศาสตราจารย์ ดร. ชิตชนก เหลือสินทรัพย์ และผู้ช่วยศาสตราจารย์ ดร. กิติพร พลายมาศ กรรมการสอบโครงการสำหรับข้อเสนอแนะต่าง ๆ ที่ควรแก้ไข และได้ชี้แนะแนวทางให้โครงการมีความสมบูรณ์มากขึ้น

ขอขอบพระคุณ ครอบครัวที่คอยสนับสนุน ให้กำลังใจและเป็นแรงผลักดัน จนโครงการชิ้นนี้สามารถสำเร็จลุล่วงไปได้

สุดท้ายนี้ขอขอบพระคุณอาจารย์ท่านอื่นที่ไม่ได้กล่าวนามไว้ ณ ที่นี้ ที่ได้ถ่ายทอดความรู้ให้ ผู้วิจัยมีความรู้ ความเข้าใจในทฤษฎีต่าง ๆ และได้ฝึกทักษะให้ผู้วิจัยมีความสามารถในการแก้ปัญหาที่ซับซ้อน อันเป็นประโยชน์ในการดำเนินโครงการนี้

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	ง
บทคัดย่อภาษาอังกฤษ .....	จ
กิตติกรรมประกาศ .....	ฉ
สารบัญ .....	ช
สารบัญตาราง .....	ฌ
สารบัญภาพ .....	ญ
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและเหตุผลการวิจัย .....	1
1.2 วัตถุประสงค์ของการวิจัย .....	2
1.3 ขอบเขตการวิจัย .....	2
1.4 ขั้นตอนการวิจัย .....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	4
1.6 โครงสร้างของรายงาน .....	4
บทที่ 2 งานวิจัยที่เกี่ยวข้อง .....	5
2.1 หลักการและทฤษฎีที่เกี่ยวข้อง .....	5
2.2 ทบทวนงานวิจัยที่เกี่ยวข้อง .....	9
บทที่ 3 วิธีการวิจัย .....	11
3.1 บทนำ .....	12
3.2 การสร้างสัญญาณ .....	12
3.3 การสร้างกราฟ .....	13
3.4 การปรับสัญญาณกราฟเรียบแบบอุดมคติ .....	15



3.5 การกำหนดค่า magnitude ประจำอำเภอสำหรับนำไประบุรูปแบบการระบาด.....	15
บทที่ 4 ผลการวิจัย .....	21
4.1 ผลการทดลองการปรับสัญญาณเรียงด้วย filter ที่แตกต่างกัน.....	21
4.2 การทดสอบสมมติฐาน T-test.....	23
4.3 การสร้าง Heatmap ที่ได้จากค่า magnitude ประจำอำเภอ .....	31
บทที่ 5 ข้อเสนอแนะ.....	35
5.1 ข้อเสนอแนะ.....	35
5.2 ข้อเสนอแนะ .....	35
รายการอ้างอิง.....	37
ภาคผนวก ก แบบเสนอหัวข้อโครงการ รายวิชา 2301399 Project Proposal ปีการศึกษา 2564 .	40
ภาคผนวก ข ภาพผลลัพธ์สัญญาณที่ถูกปรับให้เรียบที่ได้ในแต่ละ filter.....	46
ภาคผนวก ค ตัวอย่างส่วนของโปรแกรมที่ใช้ในการประมวลผลสัญญาณกราฟ.....	53
ประวัติผู้เขียน .....	60

## สารบัญตาราง

หน้า

ตารางที่ 3.1: ตัวอย่างค่าสัญญาณ แถวคืออำเภอ 928 อำเภอ .....	13
ตารางที่ 3.2: ตัวอย่างค่า magnitude ทั้ง 39 สัปดาห์ ในแต่ละอำเภอ.....	16
ตารางที่ 3.3: ตัวอย่างค่า magnitude สัปดาห์ที่ 13 และ 14 ในแต่ละอำเภอ .....	16
ตารางที่ 3.4: ตัวอย่างค่า magnitude ในแต่ละอำเภอโดยมีอำเภอทั้งหมด 928 อำเภอ ที่คำนวณได้จากสมการ (8).....	18
ตารางที่ 3.5: ตัวอย่างค่า magnitude รายสัปดาห์ 39 สัปดาห์ในแต่ละอำเภอโดยมีอำเภอทั้งหมด 928 อำเภอ ที่คำนวณได้จากสมการ (9).....	19
ตารางที่ 4.1: ความแปรปรวนทั้งหมดของสัญญาณที่ผ่านการกรองด้วย filter ที่ต่างกันต่อความแปรปรวนทั้งหมดของสัญญาณตั้งต้น .....	21
ตารางที่ 4.2: การแจกแจงความถี่ของค่า magnitude ที่สร้างได้จาก filter ที่ต่างกัน .....	22
ตารางที่ 4.3: ตัวอย่างกลุ่มข้อมูลที่นำมาทดสอบ .....	24
ตารางที่ 4.4: ตัวอย่างค่าความแปรปรวนประจำอำเภอในแต่ละกลุ่มข้อมูลที่นำมาทดสอบ.....	27

## สารบัญภาพ

หน้า

ภาพที่ 2.1: สัญญาณกราฟจากข้อมูลอุณหภูมิในแต่ละบริเวณพื้นที่.....	6
ภาพที่ 2.2: สัญญาณกราฟที่ผ่านการปรับให้เรียบ .....	9
ภาพที่ 3.1: ขั้นตอนวิธีในการประมวลสัญญาณกราฟ .....	11
ภาพที่ 3.2: สัญญาณของจำนวนผู้ติดเชื้อรายสัปดาห์ในแต่ละอำเภอ.....	13
ภาพที่ 3.3: ตำแหน่งของอำเภอในบริเวณจังหวัดกรุงเทพมหานคร.....	14
ภาพที่ 3.4: จุดยอดที่ถูกปรับให้มีค่าใกล้เคียงกับค่าในบริเวณรอบข้าง.....	17
ภาพที่ 4.1: การแจกแจงความถี่ของค่า magnitude ของสัญญาณที่กรองจาก filter ที่กำหนด .....	23
ภาพที่ 4.2: สัญญาณที่สร้างจากข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงของกลุ่มข้อมูลที่ 1 .....	25
ภาพที่ 4.3: สัญญาณที่สร้างจากข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงของกลุ่มข้อมูลที่ 2 .....	25
ภาพที่ 4.4: สัญญาณที่สร้างจากข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงของกลุ่มข้อมูลที่ 3 .....	26
ภาพที่ 4.5: การกระจายค่าความแปรปรวนประจำอำเภอของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 โดย ผ่านการทดสอบ T-test ที่ $p \text{ value} = 0.05$ .....	28
ภาพที่ 4.6: การกระจายค่าความแปรปรวนประจำอำเภอของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 โดย ผ่านการทดสอบ T-test ที่ $p \text{ value} = 0.05$ .....	28
ภาพที่ 4.7: การกระจายจำนวนผู้ติดเชื้อเฉลี่ยของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 โดยผ่านการ ทดสอบ T-test ที่ $p \text{ value} = 0.05$ .....	29
ภาพที่ 4.8: การกระจายจำนวนผู้ติดเชื้อเฉลี่ยของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 โดยผ่านการ ทดสอบ T-test ที่ $p \text{ value} = 0.05$ .....	30
ภาพที่ 4.9: heatmap ที่สร้างจาก magnitude ประจำอำเภอ.....	31
ภาพที่ 4.10: heatmap ที่สร้างจากจำนวนผู้ติดเชื้อสะสม .....	32
ภาพที่ 4.11: heatmap ที่สร้างจาก magnitude ที่สร้างจากสัญญาณความถี่ต่ำประจำอำเภอ .....	33
ภาพที่ 4.12: heatmap ที่สร้างจาก magnitude ที่สร้างจากสัญญาณความถี่สูงประจำอำเภอ .....	34

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและเหตุผลการวิจัย

การระบาดของ COVID-19 เป็นปัญหาที่กระทบต่อประเทศไทยในทุกด้าน ซึ่งการควบคุมการระบาดได้รวดเร็วจะช่วยลดความเสียหายที่เกิดขึ้นได้อย่างมีประสิทธิภาพที่สุด จากการศึกษาของ Wijitbusaba Marome และ Rajib Shaw [1] พบว่าประเทศไทยสามารถควบคุมการระบาดของ COVID-19 ได้อย่างมีประสิทธิภาพในการระบาดระลอกแรกหรือช่วงเวลา 12 มกราคม 2563 ถึง 19 ธันวาคม 2563 และลดความเสียหายที่อาจเกิดขึ้นได้เป็นอย่างดี แต่ในการระบาดระลอกที่สามหรือตั้งแต่ 1 เมษายน 2564 ประเทศไทยไม่สามารถควบคุมการระบาดได้เหมือนการระบาดระลอกแรก รวมทั้งการกลายพันธุ์ของเชื้อที่ทำให้การระบาดเป็นไปอย่างรวดเร็วมากขึ้น ส่งผลให้การควบคุมการระบาดเป็นไปได้ยากกว่าเดิม

เพื่อให้การควบคุมการระบาดมีประสิทธิภาพมากขึ้น เครื่องมือต่างๆ ที่ช่วยในการวิเคราะห์การระบาดหรือคาดการณ์การระบาดของ COVID-19 จึงถูกพัฒนาและนำมาใช้ โดยเครื่องมือที่มีการพัฒนาขึ้นในประเทศไทยส่วนมากจะเป็นการวิเคราะห์การระบาดจากข้อมูลดิบบางอย่างที่เก็บมา เช่น การนำตัวเลขผู้ติดเชื้อไปคาดการณ์จำนวนผู้ติดเชื้อรายใหม่โดยใช้ตัวแบบจำลอง SEIR [7] หรือการนำตัวแบบจำลอง SEIR ไปใช้ในการหาความสัมพันธ์การระบาดของ COVID-19 กับอุณหภูมิ [2] ซึ่งการวิเคราะห์ที่กล่าวมาไม่ได้มีการนำข้อมูลด้านอื่นมาศึกษาวิเคราะห์เพื่อหารูปแบบการระบาดร่วมด้วย เช่น ข้อมูลรับรู้เชิงพื้นที่ (spatial awareness) หรือจำนวนประชากรในแต่ละพื้นที่

เครื่องมือที่ศึกษาและพัฒนาสำหรับช่วยในการควบคุมการระบาด COVID-19 มีด้วยกันหลายประเภท เช่น การใช้การเรียนรู้ของเครื่องในการระบุผู้ติดเชื้อจากภาพ x-ray ในปอด [3] การระบุความเสี่ยงที่จะติดเชื้อ COVID-19 จากการใช้ reinforcement learning ใน joint graph [4] ปัญหาที่เกิดขึ้นจากการนำเครื่องมือเหล่านี้มาใช้ในประเทศไทย คือข้อมูลที่มีอยู่ในประเทศไทยนำไปปรับใช้ได้ยาก เนื่องจากข้อมูลที่เกี่ยวข้องไม่ได้มีการเก็บข้อมูลไว้อย่างถูกต้องครบถ้วน หรือข้อมูลที่เกี่ยวข้องไม่มีการเผยแพร่ต่อสาธารณะหรือจำกัดการเข้าถึง ดังนั้นการสร้างเครื่องมือที่ซับซ้อนเพื่อช่วยในการควบคุมการระบาด COVID-19 ในประเทศไทยจึงเป็นไปได้ยาก

จากงานของ Yang Li และ Gonzalo Mateos [5] ที่ได้มีการประยุกต์นำการประมวลสัญญาณกราฟไปศึกษาความถี่กราฟเพื่อนำไปวิเคราะห์รูปแบบการระบาดของ COVID-19 ในประเทศสหรัฐอเมริกา เพื่อระบุว่าบริเวณใดมีการระบาดที่ต่อเนื่องกันเป็นรูปแบบเดียวกันเมื่อเทียบกับบริเวณ

รอบข้าง (low-pass regions) และบริเวณใดมีการระบาดในลักษณะที่แตกต่างกันเมื่อเทียบกับ บริเวณรอบข้าง (high-pass regions) ซึ่งทางผู้วิจัยเห็นว่าผลที่ได้จากการวิเคราะห์ข้อมูลที่หลากหลาย แสดงให้เห็นถึงการระบาด COVID-19 ในเชิงลึกมากกว่าผลที่ได้จากการวิเคราะห์เพียงข้อมูลผู้ติดเชื้อ รายวันอย่างเดียว ซึ่งผลลัพธ์เชิงลึกนี้สามารถนำไปประยุกต์เป็นข้อมูลเพิ่มเติมสำหรับการ ประกอบการตัดสินใจในการวางแผนควบคุมการระบาด ทั้งยังสามารถนำไปปรับพัฒนาให้ใช้ได้จริง ในประเทศไทยอีกด้วย

จากการศึกษาในข้างต้น ผู้วิจัยมีความสนใจในการนำงานของ Yang Li และ Gonzalo Mateos มาต่อยอดเพื่อพัฒนาเครื่องมือสำหรับวิเคราะห์การระบาด COVID-19 ในประเทศไทย อย่างไรก็ตามงานของ Yang Li และ Gonzalo Mateos ได้มีการนำเอาข้อมูล สถานที่ทำงานของ ประชากรในแต่ละอำเภอ (commute flow) มาสร้างเป็นกราฟสำหรับนำเอาไปวิเคราะห์ความถี่ด้วย ซึ่งข้อมูลในส่วนนี้ไม่ได้มีการจัดเก็บไว้หรือมีการเผยแพร่ให้เป็นสาธารณะในประเทศไทย ทำให้ผู้วิจัย ต้องหาวิธีการสร้างกราฟแบบอื่นแทน จากการศึกษาในงานของ Leah Goldsberry และคณะ [8] พบว่ามีการนำข้อมูลระยะห่างระหว่างจุดยอดมาสร้างเป็นกราฟ ทางผู้วิจัยจึงมีความสนใจที่จะนำ ข้อมูลเชิงพื้นที่คือ ระยะห่างระหว่างอำเภอ มาสร้างเป็นกราฟแทนการใช้ข้อมูล commute flow นอกจากนี้ผลลัพธ์ที่ได้จากงานของ Yang Li และ Gonzalo Mateos มีค่าผลลัพธ์อยู่ 2 ชุด คือ magnitude ความถี่ต่ำ และ magnitude ความถี่สูง ที่แสดงถึงรูปแบบการระบาด โดยผลลัพธ์นี้อยู่ ในช่วงค่าที่แตกต่างกัน ทำให้การระบุรูปแบบการระบาดในบางอำเภอทำได้ไม่ชัดเจน เช่น อำเภอที่ทั้ง magnitude ความถี่ต่ำ และ magnitude ความถี่สูงมีค่าสูงทั้งคู่ อำเภอดังกล่าวจะไม่สามารถระบุได้ อย่างชัดเจนว่ามีรูปแบบการระบาดแบบใดผู้วิจัยจึงมีความสนใจที่จะปรับปรุงการกำหนดค่าผลลัพธ์ที่ สามารถระบุรูปแบบการระบาดในแต่ละอำเภอให้มีความชัดเจนมากขึ้น

## 1.2 วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาเครื่องมือในการระบุรูปแบบการระบาดของ COVID-19 ในประเทศไทย โดยสร้าง กราฟจากข้อมูลระยะห่างระหว่างอำเภอ วิเคราะห์ความถี่กราฟ และนำผลที่ได้มาจัดการกับข้อมูลผู้ติด เชื้อรายสัปดาห์ในแต่ละอำเภอ เพื่อระบุว่าในบริเวณใกล้เคียงกัน บริเวณใดมีจำนวนผู้ติดเชื้อใกล้เคียง กัน และบริเวณใดมีจำนวนผู้ติดเชื้อแตกต่างกัน

## 1.3 ขอบเขตการวิจัย

โครงงานนี้ใช้ข้อมูลผู้ติดเชื้อรายสัปดาห์ในแต่ละอำเภอของประเทศไทย มีทั้งหมด 928 อำเภอ จาก 77 จังหวัด โดยนำข้อมูลมาจากเว็บไซต์ data.go.th ใช้ข้อมูลตั้งแต่เริ่มการระบาดระลอก ที่ 3 จนถึงช่วงการระบาดสงบลง หรือข้อมูลตั้งแต่วันที่ 1 เมษายน 2564 จนถึงวันที่ 30 ธันวาคม 2564 เป็นระยะเวลา 39 สัปดาห์ ผลลัพธ์ที่ได้จะเป็นข้อมูลชุดเดียวที่แสดงถึงรูปแบบการติดเชื้อ 3

แบบตลอด 39 สัปดาห์ คือ บริเวณที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับรอบข้าง บริเวณที่มีจำนวนผู้ติดเชื้อน้อยกว่ารอบข้างและบริเวณที่มีจำนวนผู้ติดเชื้อมากกว่ารอบข้าง

## 1.4 ขั้นตอนการวิจัย

ก. แผนการดำเนินงาน

1. ศึกษาค้นคว้าทฤษฎีที่เกี่ยวข้อง
2. เตรียมข้อมูลผู้ติดเชื้อรายสัปดาห์ในแต่ละอำเภอ
3. สร้างกราฟจากข้อมูลระยะห่างระหว่างอำเภอ
4. ปรับสัญญาณกราฟให้เรียบแบบในอุดมคติ
  - 4.1 ทำ graph Fourier transform เพื่อวิเคราะห์ความถี่ของกราฟ
  - 4.2 สร้าง ideal low-pass filter จากการเลือก eigenvector มาตามปริมาณที่กำหนด
  - 4.3 ทำการ filter สัญญาณ เพื่อปรับสัญญาณกราฟให้เรียบ
  - 4.4 หา magnitude ของแต่ละอำเภอ
5. ประเมินผลลัพธ์ที่ได้จากการปรับสัญญาณกราฟให้เรียบแบบในอุดมคติ
6. สร้าง heat map สำหรับดูการระบาดของ COVID-19
  - 6.1 สร้าง heatmap จากข้อมูลผู้ติดเชื้อรายสัปดาห์ทั้ง 39 สัปดาห์
  - 6.2 สร้าง heatmap จากข้อมูลผู้ติดเชื้อรายสัปดาห์โดยแบ่งสร้างทีละ 1 สัปดาห์
7. วิเคราะห์และอภิปรายผล
8. จัดทำเอกสาร

ข. ระยะเวลาที่ศึกษา

ขั้นตอนการดำเนินงาน	ปี 2564						ปี 2565		
	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1. ศึกษาค้นคว้าทฤษฎีที่เกี่ยวข้อง									
2. เตรียมข้อมูลผู้ติดเชื้อราย สัปดาห์ในแต่ละอำเภอ									
3. สร้างกราฟจากข้อมูลระยะห่าง ระหว่างอำเภอ									
4. ปรับสัญญาณกราฟให้เรียบแบบ ในอุดมคติ									

5. ประเมินผลลัพธ์ที่ได้จากการปรับสัญญาณกราฟให้เรียบแบบในอุดมคติ									
6. สร้าง heat map สำหรับดูการระบาดของ COVID-19									
7. วิเคราะห์และอภิปรายผล									
8. จัดทำเอกสาร									

### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ต่อผู้พัฒนา

1. ฝึกฝนและพัฒนาทักษะในการวางแผนและการทำงานเป็นขั้นตอน
2. ฝึกฝนและพัฒนาทักษะในการพัฒนาเครื่องมือสำหรับการปรับสัญญาณกราฟให้เรียบแบบในอุดมคติ

ประโยชน์ที่ได้จากโครงการ

1. ผู้ใช้ได้ข้อมูลเชิงลึกเกี่ยวกับรูปแบบการระบาดของ COVID-19 นอกเหนือจากแผนภาพจำนวนผู้ติดเชื้อรายสัปดาห์
2. ผู้ใช้สามารถนำข้อมูลรูปแบบการระบาดเชิงพื้นที่ไปประยุกต์ใช้ในการวางแผนควบคุมการระบาดได้ดียิ่งขึ้น เช่น การใช้เป็นเกณฑ์ประกอบในการแบ่งพื้นที่ความรุนแรงของการระบาด
3. สามารถนำผลการสร้างกราฟไปประยุกต์ต่อยอดในการพัฒนาเครื่องมืออื่นที่มีความใกล้เคียงกันได้

### 1.6 โครงสร้างของรายงาน

บทที่ 2 จะกล่าวถึงหลักการและงานวิจัยที่เกี่ยวข้อง

บทที่ 3 จะกล่าวถึงวิธีการวิจัยในการเสนอแนวทางการปรับปรุงประมวลผลสัญญาณกราฟ การสร้างกราฟ การปรับสัญญาณกราฟเรียบแบบในอุดมคติ การกำหนดค่า magnitude ประจำอำเภอ สำหรับนำไประบุรูปแบบการระบาดของ COVID-19 ในประเทศไทย

บทที่ 4 จะกล่าวถึงผลการดำเนินงานวิจัยของการประมวลผลสัญญาณกราฟ

บทที่ 5 จะกล่าวถึงการสรุปผลวิจัยในการประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย และข้อเสนอแนะ

## บทที่ 2

### งานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการและทฤษฎีรวมถึงงานวิจัยที่เกี่ยวข้องกับการประมวลผลสัญญาณกราฟ

#### 2.1 หลักการและทฤษฎีที่เกี่ยวข้อง

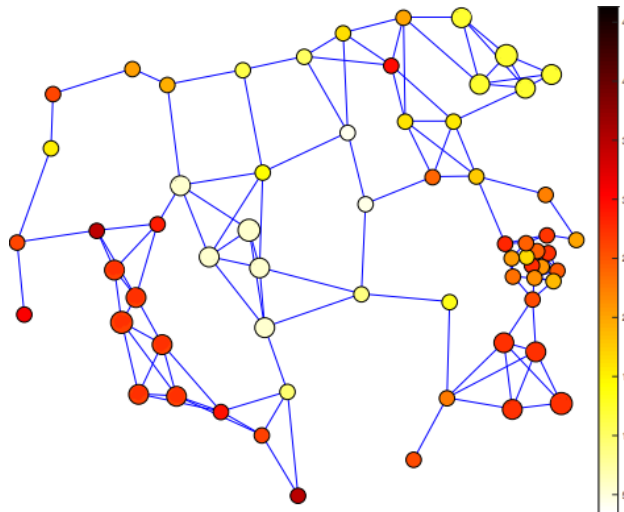
##### 2.1.1 การประมวลผลสัญญาณกราฟ

การประมวลผลสัญญาณกราฟ [6], [9] เป็นวิธีการวิจัยที่ผู้คนเริ่มให้ความสนใจในช่วงไม่กี่ปีที่ผ่านมาส่งผลให้เกิดการแก้ปัญหาขั้นสูงที่แปลกใหม่ และมีการประยุกต์ใช้งานที่หลากหลาย โดยโดเมนของสัญญาณมักจะไม่ใช่กลุ่มข้อมูลที่มีเวลาหรือพื้นที่ห่างกันเท่ากัน แต่จะเกี่ยวข้องกับคุณสมบัติอื่นๆ ของระบบหรือเครือข่ายที่พิจารณา กล่าวคือโดเมนของสัญญาณจะถูกนิยามขึ้นโดยกราฟ กราฟจะประกอบไปด้วยจุดยอด และเส้นเชื่อมระหว่างจุดยอด โดยจุดยอดจะเก็บค่าข้อมูลที่ได้จากข้อมูลสัญญาณ และเส้นเชื่อมระหว่างจุดยอดจะแสดงถึงความสัมพันธ์ระหว่างจุดยอด โดยหากระหว่างจุดยอดไม่มีความเกี่ยวข้องกันจะมีน้ำหนักเป็น 0 หรือไม่มีเส้นเชื่อมระหว่างจุด แต่หากมีความเกี่ยวข้องกันมากก็จะมีค่ามาก ซึ่งการสร้างกราฟในลักษณะนี้จะทำให้การประมวลผลสัญญาณกราฟสามารถที่จะมีข้อมูลที่เกี่ยวข้องกับคุณสมบัติของสัญญาณเข้ามารวมประมวลผลด้วยซึ่งแตกต่างจากการประมวลผลสัญญาณแบบปกติ

##### 2.1.2 การสร้างกราฟสำหรับการประมวลผลสัญญาณกราฟ

โดยส่วนมากแล้วข้อมูลสัญญาณจะเป็นข้อมูลที่มีการกำหนดเอาไว้อย่างชัดเจนแล้ว แต่ข้อมูลกราฟซึ่งแสดงถึงความสัมพันธ์ระหว่างสัญญาณมักจะไม่ได้ถูกกำหนดเอาไว้อย่างชัดเจน [6], [9] โดยข้อมูลที่น่ามาสร้างกราฟจะต้องพิจารณาตามคุณสมบัติที่เกี่ยวข้องกับสมมติฐานที่ตั้งไว้สำหรับความสัมพันธ์ระหว่างสัญญาณ เช่น การนำสัญญาณกราฟไปพิจารณาอุณหภูมิในพื้นที่ต่างๆ [9] โดยตั้งสมมติฐานว่าบริเวณใกล้เคียงกันจะมีอุณหภูมิใกล้เคียงกัน กล่าวคือจุดยอดที่อยู่ใกล้กันจะมีอุณหภูมิใกล้เคียงกัน จะได้ว่าสัญญาณคือข้อมูลอุณหภูมิ จุดยอดคือตำแหน่งที่ทำการเก็บข้อมูลอุณหภูมิ และเส้นเชื่อมคือระยะห่างระหว่างจุดยอด ดังภาพที่ 2.1





ภาพที่ 2.1: สัญญาณกราฟจากข้อมูลอุณหภูมิในแต่ละบริเวณพื้นที่

#### 2.1.2.1 การสร้างกราฟจากตำแหน่งของจุดยอด

สำหรับกราฟที่กำหนดด้วยข้อมูลตำแหน่งของสัญญาณ จะมีสมมติฐานที่ตั้งว่าสัญญาณที่อยู่ใกล้กันจะมีค่าใกล้เคียงกัน จะได้จุดยอดคือตำแหน่งของสัญญาณ และกำหนดน้ำหนักของเส้นเชื่อมจากระยะห่างระหว่างจุด โดยจุดที่ห่างกันจะมีน้ำหนักน้อยกว่าจุดที่อยู่ใกล้กัน วิธีหนึ่งที่ใช้กันอย่างแพร่หลายคือคำนวณน้ำหนักของเส้นเชื่อมจาก Gaussian function [6] พิจารณาจุดยอด  $n$  และ  $m$  ซึ่งตำแหน่งของจุดยอดถูกกำหนดไว้อย่างชัดเจน

$$W_{nm} = \begin{cases} e^{\frac{-r_{nm}^2}{\tau^2}} & , r_{nm} < \kappa \\ 0 & , r_{nm} \geq \kappa \end{cases} \quad (1)$$

โดยที่  $W_{nm}$  เป็นน้ำหนักของเส้นเชื่อมระหว่างจุดยอด  $n, m$

$r_{nm}$  เป็นระยะห่าง Euclid ระหว่างจุดยอด  $n, m$

$\tau$  และ  $\kappa$  เป็นค่าคงตัว

โดยสมการ (1) จะมีการปรับค่าน้ำหนักตามระยะห่าง ระยะห่างที่มากขึ้นจะถูกปรับลงอย่าง exponential โดยจะมีการปรับตามพารามิเตอร์ 2 ตัวคือ  $\tau$  และ  $\kappa$  โดย  $\tau$  คือ พารามิเตอร์สำหรับการปรับ Gaussian function และ  $\kappa$  คือพารามิเตอร์สำหรับเป็นเกณฑ์ในการการเลือก

ระยะห่างที่นำมาถูกปรับ ซึ่งระยะห่างที่เกิน  $K$  จะไม่ถูกนำมาพิจารณาเป็นเส้นเชื่อม กราฟที่ได้จากสมการ (1) จะเป็นกราฟสมมาตรไม่มีทิศทาง

### 2.1.3 Graph Fourier transform จากกราฟ Laplacian

กระบวนการนี้จะเป็นการแยกส่วนประกอบของสัญญาณกราฟให้อยู่ในรูปของส่วนประกอบเชิงตั้งฉาก (orthonormal components) ซึ่งจะสามารถแสดงถึงความแปรปรวนของสัญญาณกราฟจากลักษณะของกราฟ [5] โดย graph Fourier transform จะสามารถแสดงสัญญาณกราฟให้อยู่ใน 2 โดเมนที่แตกต่างกันคือ โดเมนของจุดยอด และโดเมนความถี่กราฟซึ่งแสดงด้วย spectral basis ของกราฟ กระบวนการนี้จะทำให้สามารถจัดการกับสัญญาณในโดเมนของความถี่ได้ ซึ่งโดเมนความถี่นี้สามารถใช้เป็นเครื่องมือสำหรับการวิเคราะห์ความสัมพันธ์ของสัญญาณในบริเวณที่ใกล้เคียงกันด้วยกราฟ

พิจารณาสัญญาณบนจุดยอดของกราฟซึ่งเป็นกราฟมีน้ำหนัก ไม่มีทิศทาง และเป็นกราฟเชื่อมโยง จะมีสัญญาณ  $\mathbf{x} = [x_1, \dots, x_N]^T \in \mathbb{R}^N$  และกราฟมีน้ำหนัก  $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{W})$  โดยที่  $\mathcal{V} = \{1, \dots, N\}$  คือเซตของจุดยอด  $N$  จุดหรือก็คือจุดยอดที่แทนถึงสัญญาณ และ  $\mathbf{W} : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}_+$  คือน้ำหนักของเส้นเชื่อมของกราฟที่แสดงถึงความสัมพันธ์ระหว่างสัญญาณ และเนื่องจากเป็นกราฟไม่มีทิศทางหรือก็คือเป็นกราฟสมมาตรจึงได้ว่า  $w_{ij} = w_{ji}$  สำหรับทุกเส้นเชื่อม  $(i, j)$  โดยที่  $(i, j) \in \mathcal{E}$  จะมีดีกรีเมทริกซ์  $\mathbf{D} \in \mathbb{R}_+^{n \times n}$  คือเมทริกซ์แนวทแยงซึ่งสมาชิกตัวที่  $i$  จะมีค่าคือ  $D_{ii} = \sum_{j=1}^n w_{ij}$  และเมทริกซ์ Laplacian  $\mathbf{L}$  นิยามโดย  $\mathbf{L} = \mathbf{D} - \mathbf{W} \in \mathbb{R}^{n \times n}$

พิจารณาถึงการแยกส่วนประกอบ eigenvector ของกราฟ Laplacian  $\mathbf{L}$  จะมีเมทริกซ์  $\mathbf{A} = \text{diag}(\lambda_1, \dots, \lambda_n)$  ซึ่งเป็นเมทริกซ์แนวทแยงมี Laplacian eigenvalue ที่ไม่เป็นจำนวนลบ และมี  $\mathbf{V} = [v_1, \dots, v_n]$  เป็นเมทริกซ์ของ eigenvectors ที่ตั้งฉากกัน โดยเมทริกซ์ Laplacian  $\mathbf{L}$  จะสามารถแยกส่วนประกอบได้เป็น  $\mathbf{L} = \mathbf{V}\mathbf{A}\mathbf{V}^T$  graph Fourier transform ของสัญญาณ  $\mathbf{x}$  จาก Laplacian  $\mathbf{L}$  คือ  $\tilde{\mathbf{x}} = \mathbf{V}^T \mathbf{x}$  และ inverse graph Fourier transform ของ  $\tilde{\mathbf{x}}$  คือ  $\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}}$

### 2.1.4 Graph Fourier transform กับความเรียบของสัญญาณ

คุณลักษณะที่สำคัญที่ได้จากกระบวนการ graph Fourier transform [5] คือ ทำให้ความแปรปรวนของสัญญาณกราฟถูกจัดรูปให้อยู่ในลักษณะเดียวกันกับความถี่ในการวิเคราะห์ Fourier ของ temporal signals

นิยามความแปรปรวนทั้งหมด (total variation) ของสัญญาณ  $TV(\mathbf{x})$  และ Laplacian  $\mathbf{L}$  ซึ่งสร้างจากกราฟของสัญญาณ

$$TV(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{i \neq j} W_{ij} (x_i - x_j)^2 \quad (2)$$

ความแปรปรวนทั้งหมดเปรียบเสมือนตัววัดความเรียบของสัญญาณ ซึ่งจะบ่งบอกถึงปริมาณที่สัญญาณ  $\mathbf{x}$  แตกต่างกันมากน้อยเพียงใด โดยพิจารณาร่วมกับความสัมพันธ์ระหว่างสัญญาณตามสมมติฐานที่ตั้งขึ้นซึ่งคือเมทริกซ์น้ำหนัก  $\mathbf{W}$

จากนิยามความแปรปรวนทั้งหมด จะได้ว่าความแปรปรวนทั้งหมดของ eigenvectors  $\mathbf{v}_k$  หรือ  $TV(\mathbf{v}_k) = \mathbf{v}_k^T \mathbf{L} \mathbf{v}_k = \lambda_k$  ทำให้สามารถสรุปได้ว่า eigenvalues  $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_N$  สามารถพิจารณาได้เป็นความถี่กราฟ

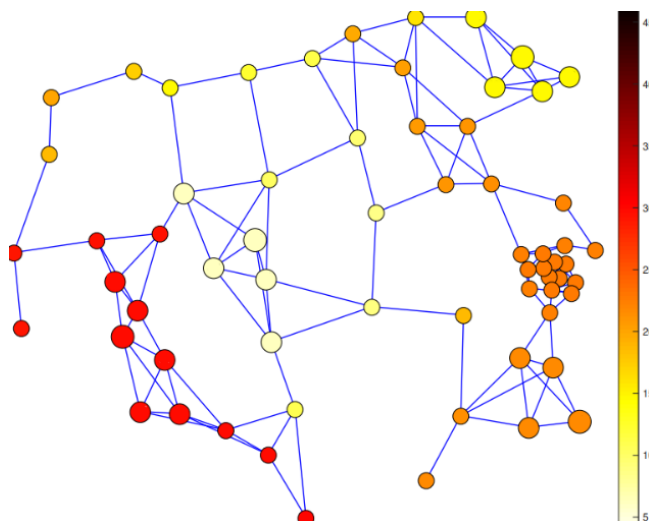
### 2.1.5 กระบวนการปรับสัญญาณกราฟให้เรียบแบบอุดมคติ

จากที่ eigenvalues สามารถพิจารณาเป็นความถี่ของกราฟ การสร้าง ideal low-pass filter [5], [6], [8] สำหรับกรองสัญญาณกราฟในโดเมนความถี่สามารถทำได้โดยการเลือก eigenvectors ที่มีค่า eigenvalues ไม่เกิน  $\lambda_c$  ตามสมการ (3) หรือสร้างจากการเลือก eigenvectors มาตามปริมาณที่กำหนด  $N_L$  ตามสมการ (4) จะได้ ideal low-pass filter  $\mathbf{H}_L = \mathbf{V} \widetilde{\mathbf{H}}_L \mathbf{V}^T$  โดยที่  $\widetilde{\mathbf{H}}_L = \text{diag}(\widetilde{h}_{Lk})$  เป็นเมทริกซ์แนวทแยงสมาชิกจะมีค่าเป็น 0 หรือ 1 ตามข้อพิจารณาการที่ eigenvector จะถูกเลือก โดยปริมาณ eigenvector ที่ถูกเลือกจะแสดงถึงความหนักของการปรับสัญญาณให้เรียบ โดยหากเลือกให้ eigenvector มีปริมาณน้อย สัญญาณจะถูกปรับให้เรียบมาก แต่หากเลือกให้ eigenvector มีปริมาณมาก สัญญาณจะถูกปรับให้เรียบน้อย หรือเทียบเท่ากับสัญญาณดั้งเดิม ได้สัญญาณที่ผ่านการกรอง  $\mathbf{x}_L = \mathbf{H}_L \mathbf{x} = \mathbf{V} \widetilde{\mathbf{H}}_L \mathbf{V}^T \mathbf{x}$

$$\widetilde{h}_{Lk} = \begin{cases} 1, & \lambda_k < \lambda_c \\ 0, & \lambda_k \geq \lambda_c \end{cases} \quad (3)$$

$$\widetilde{h}_{Lk} = \mathbb{I}\{k < N_L\} \quad (4)$$

พิจารณา filter  $\mathbf{H}_L = \mathbf{V} \widetilde{\mathbf{H}}_L \mathbf{V}^T$  สามารถจัดรูปให้อยู่ใน Laplacian powers ได้เป็น  $\mathbf{H}_L = \sum_{k=0}^{n-1} h_{Lk} \mathbf{L}^k$  โดย  $\mathbf{L}^k \mathbf{x}$  คือการเฉลี่ยของค่าในจุดยอดกับจุดยอดที่อยู่ห่างออกไป  $k$ -hop ดังนั้นสัญญาณที่ผ่านการกรอง  $\mathbf{x}_L = \mathbf{H}_L \mathbf{x} = \sum_{k=0}^{n-1} h_{Lk} \mathbf{L}^k \mathbf{x}$  จะเปรียบเสมือนผลลัพธ์ที่ได้จากการทำ localized averaging ของในแต่ละสมาชิก  $\mathbf{x}$  ส่งผลให้  $\mathbf{x}_L$  มีความแปรปรวนทั้งหมดน้อยกว่า  $\mathbf{x}$  [8] ตัวอย่างเช่น ภาพที่ 2.2 เป็นภาพผลลัพธ์ที่ได้จากการปรับสัญญาณกราฟให้เรียบ เมื่อเทียบกับภาพที่ 2.1 จะเห็นได้ว่าในจุดบริเวณใกล้เคียงกันจะมีสีใกล้เคียงกันซึ่งอยู่ในลักษณะเดียวกันกับการทำ localized averaging



ภาพที่ 2.2: สัญญาณกราฟที่ผ่านการปรับให้เรียบ

## 2.2 ทบทวนงานวิจัยที่เกี่ยวข้อง

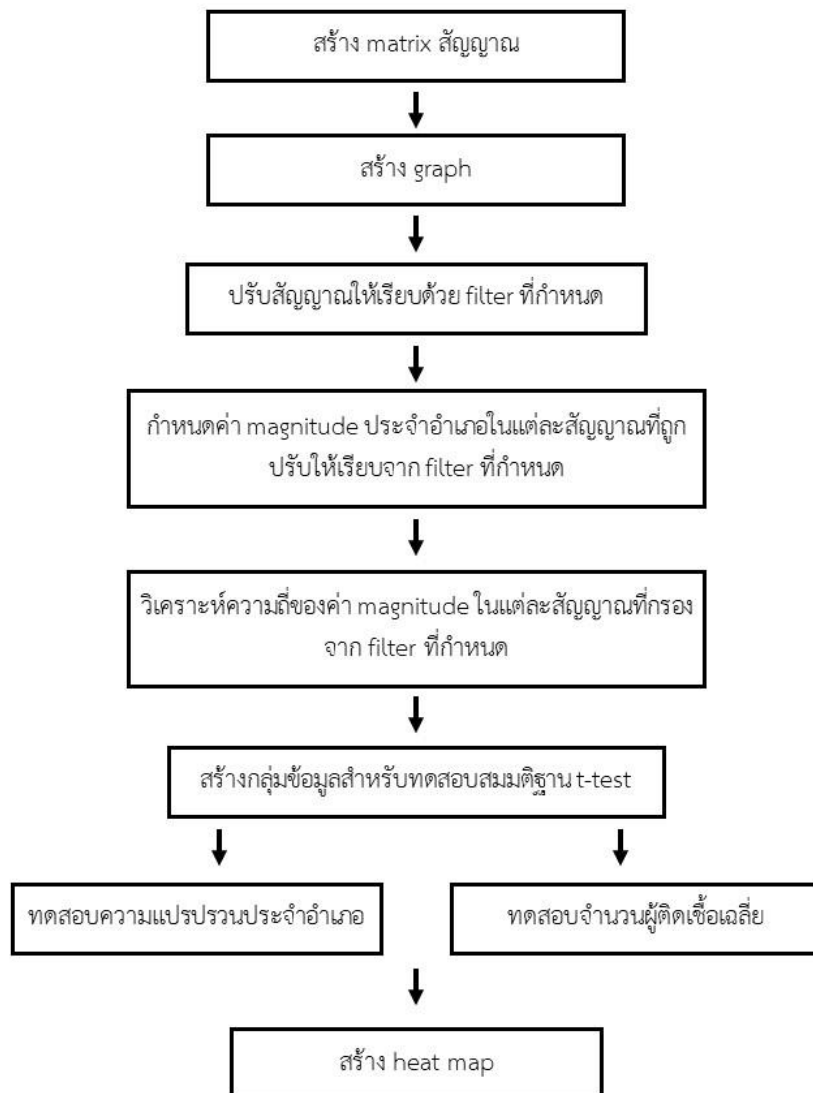
งานวิจัยของ Yang Li และ Gonzalo Mateos [5] ได้ทำการศึกษาความถี่กราฟสำหรับนำไประบุรูปแบบการระบาด COVID-19 ในสหรัฐอเมริกา มีกระบวนการวิจัยเริ่มจากนำข้อมูลจำนวนผู้ติดเชื้อ COVID-19 รายวันในระดับอำเภอมาสร้างเป็นสัญญาณ และสร้างกราฟจากข้อมูล commute flow ซึ่งเป็นข้อมูลสถานที่ทำงานของประชากรในแต่ละอำเภอ นำจำนวนประชากรมา normalize ให้เป็นค่าน้ำหนักในช่วงค่า  $[0,1]$  จากนั้นทำการวิเคราะห์ความถี่กราฟ เริ่มจากแปลงกราฟ commute flow ให้เป็นกราฟ Laplacian ทำ graph Fourier transform ได้ผลลัพธ์คือ eigenvector และ eigenvalue ซึ่งแสดงถึงความถี่กราฟ จากนั้นทำการแบ่งช่วงความถี่ของ graph signal จากการทำ graph filtering เริ่มจากการสร้าง filter มีด้วยกัน 2 แบบคือ ความถี่ต่ำ (low-pass filter) และความถี่สูง (high-pass filter) ซึ่งสร้างจากการเลือก eigenvector ที่มีค่า eigenvalue อยู่ในช่วง 20% ที่น้อยที่สุดและช่วง 20% ที่มากที่สุดตามลำดับ หลังจากนั้นทำการกรองสัญญาณด้วย filter ที่สร้าง จะได้ผลลัพธ์เป็นสัญญาณที่ถูกแบ่งออกมาอยู่ในช่วงความถี่ที่กำหนด นั่นคือ สัญญาณความถี่ต่ำ (low-pass signal) หรือสัญญาณที่มีการเปลี่ยนแปลงน้อยเมื่อเทียบกับจุดยอดในบริเวณรอบข้าง และสัญญาณความถี่สูง (high-pass signal) หรือสัญญาณที่มีการเปลี่ยนแปลงมากเมื่อเทียบกับจุดยอดในบริเวณรอบข้าง จากนั้นนำสัญญาณที่ผ่านการกรองมาคำนวณหาค่า magnitude ของแต่ละอำเภอซึ่งแสดงถึงความเป็น low-pass และ high-pass โดยรวมของอำเภอนั้นๆ โดยคำนวณมาจากค่าเฉลี่ยของค่าสมบูรณ์ของสัญญาณที่ผ่านการกรองในแต่ละอำเภอ ได้ค่า magnitude ออกมาเป็น 2 ชุด คือ magnitude ความถี่ต่ำ (low-pass magnitude) และ magnitude ความถี่สูง (high-pass magnitude) โดยค่า magnitude นี้เป็นค่าที่แสดงถึงรูปแบบการระบาดของอำเภอ กลุ่มอำเภอที่ magnitude ความถี่ต่ำมีค่าสูงแสดงถึงกลุ่มอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกันเมื่อเทียบกับบริเวณรอบข้าง ซึ่งก็คือบริเวณอำเภอที่มีการระบาดอยู่ในระดับแพร่กระจายไปอำเภอรอบข้าง และกลุ่มอำเภอที่

magnitude ความถี่สูงมีค่าสูงจะแสดงถึงกลุ่มอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างกันเมื่อเทียบกับบริเวณรอบข้าง ซึ่งก็คืออำเภอที่มีการระบาดอยู่ในระดับแพร่กระจายเพียงในอำเภอเท่านั้นยังไม่แพร่กระจายไปยังอำเภอรอบข้าง เนื่องจากค่า magnitude ที่ใช้ในการระบุรูปแบบการระบาดตั้งอยู่ในค่า 2 ชุดคือ magnitude ความถี่ต่ำ และ magnitude ความถี่สูง ทำให้การสรุปรูปแบบการระบาดในบางอำเภอทำได้ไม่ชัดเจน เช่น อำเภอที่ทั้ง magnitude ความถี่ต่ำมีค่าสูง และ magnitude ความถี่สูงมีค่าสูงเหมือนกัน ทำให้อาเภอดังกล่าวไม่สามารถระบุรูปแบบการระบาดได้อย่างชัดเจนว่าเป็นการระบาดแบบมีจำนวนผู้ติดเชื้อใกล้เคียงกับรอบข้าง หรือแตกต่างจากบริเวณรอบข้าง

### บทที่ 3

#### วิธีการวิจัย

ในบทนี้จะกล่าวถึงขั้นตอนวิธีการวิจัยการเสนอแนวทางการปรับปรุงประมวลผลสัญญาณกราฟ การสร้างสัญญาณ การสร้างกราฟ การปรับสัญญาณกราฟเรียบแบบในอุดมคติ และการกำหนดค่า magnitude สำหรับนำไประบุรูปแบบการระบาดของ COVID-19 ในประเทศไทย ดังภาพที่ 3.1



ภาพที่ 3.1: ขั้นตอนวิธีในการประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของ COVID-19 ในประเทศไทย

### 3.1 บทนำ

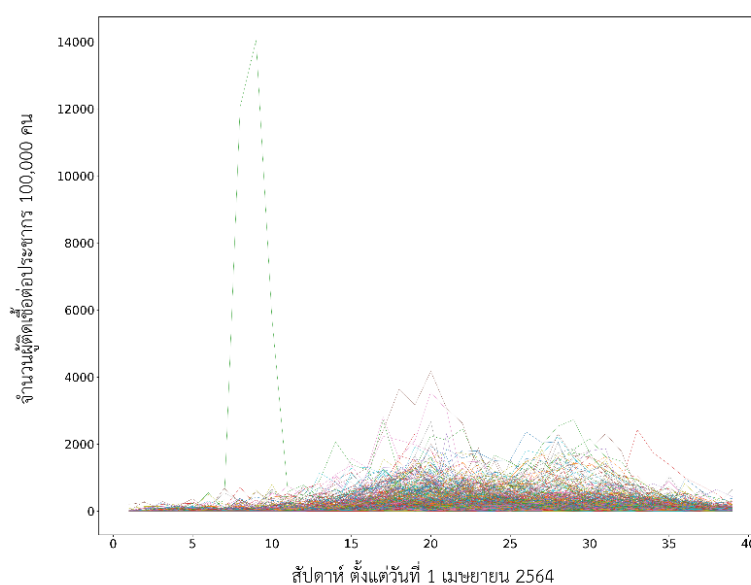
วิธีการของ Yang Li และ Gonzalo Mateos มีการพิจารณาสัญญาณในความถี่ 2 ความถี่ ได้แก่ ความถี่ต่ำ และความถี่สูง เพื่อที่จะนำมาสร้างผลลัพธ์ ส่งผลให้ได้ค่าที่อธิบายถึงรูปแบบการระบาดหรือ magnitude จำนวน 2 ชุด และช่วงค่าของ magnitude ในแต่ละสัปดาห์แตกต่างกันตามจำนวนผู้ติดเชื้อ การที่จะนำผลลัพธ์ที่ได้ไปแปลผลเพื่อสรุปรูปแบบการระบาดในแต่ละอำเภอจึงทำได้ไม่ชัดเจน จึงได้เสนอวิธีการประมวลผลสัญญาณซึ่งให้ผลลัพธ์ที่แสดงถึงการระบาดแบบต่างๆจากผลลัพธ์เพียงชุดเดียวและมีค่าอยู่ในช่วงเดียวกันในแต่ละสัปดาห์ โดยการพิจารณาสัญญาณในช่วงความถี่ต่ำเพียงอย่างเดียว จากนั้นได้เสนอวิธีการคำนวณ magnitude ประจำอำเภอจากความต่างระหว่างสัญญาณดั้งเดิมและสัญญาณที่ผ่านการกรองแทนการใช้ค่าเฉลี่ยของค่าสมบรูณ์ของสัญญาณที่ผ่านการกรอง โดยใช้หลักการที่ว่าสัญญาณที่ได้จากการกรองความถี่ต่ำเปรียบเสมือนกับการทำ localized averaging

### 3.2 การสร้างสัญญาณ

การสร้างสัญญาณจากข้อมูลจำนวนผู้ติดเชื้อรายสัปดาห์ของแต่ละอำเภอในประเทศไทย สร้างจากข้อมูลมาจากศูนย์กลางข้อมูลเปิดภาครัฐ (Open Government Data) หรือเว็บไซต์ data.go.th โดยนำข้อมูลตั้งแต่วันที่ 1 เมษายน 2564 ถึงวันที่ 30 ธันวาคม 2564 รวมทั้งสิ้นเป็นระยะเวลา 39 สัปดาห์จาก 928 อำเภอ เนื่องจากการบันทึกข้อมูลจำนวนผู้ติดเชื้อในประเทศไทยในระดับอำเภอนั้นจะมีความล่าช้ากว่าในระดับจังหวัด ทำให้ข้อมูลจำนวนผู้ติดเชื้อในแต่ละวันจะไม่ใช่อยอดจำนวนผู้ติดเชื้อที่แท้จริง เพื่อลดความผิดพลาดนี้จึงนำข้อมูลจำนวนผู้ติดเชื้อรายสัปดาห์แทนข้อมูลจำนวนผู้ติดเชื้อรายวัน จากนั้นทำการ normalize จำนวนผู้ติดเชื้อในแต่ละอำเภอด้วยการปรับเป็นจำนวนผู้ติดเชื้อในอำเภอต่อประชากรในอำเภอ 100,000 คน เพื่อให้ผลลัพธ์ที่ได้จากการระบุรูปแบบการระบาดไม่ bias กับความหนาแน่นของประชากรในแต่ละอำเภอ จากนั้นทำการสร้างเมทริกสัญญาณ  $x \in \mathbb{R}^{928 \times 39}$  โดยแถวคืออำเภอ และหลักคือจำนวนผู้ติดเชื้อในแต่ละสัปดาห์ จะได้ตัวอย่างค่าสัญญาณดังตารางที่ 3.1 และรูปสัญญาณดังภาพที่ 3.2 โดยแต่ละเส้นแทนสัญญาณของแต่ละอำเภอ

อำเภอ	สัปดาห์ที่ 1	สัปดาห์ที่ 2	...	สัปดาห์ที่ 39
1. พระนคร	0	9		122
2. ดุสิต	15	4		45
3. หนองจอก	0	1		12
4. บางรัก	9	15		133
...	...	...	...	...
925. สุโขทัย	0	1		26
926. สุโขทัย	0	2		13
927. จะนะ	0	0		13
928. จะนะ	0	0		2

ตารางที่ 3.1: ตัวอย่างค่าสัญญาณ แถวคืออำเภอ 928 อำเภอ หลักคือจำนวนผู้ติดเชื้อต่อประชากร 100,000 คนในแต่ละสัปดาห์



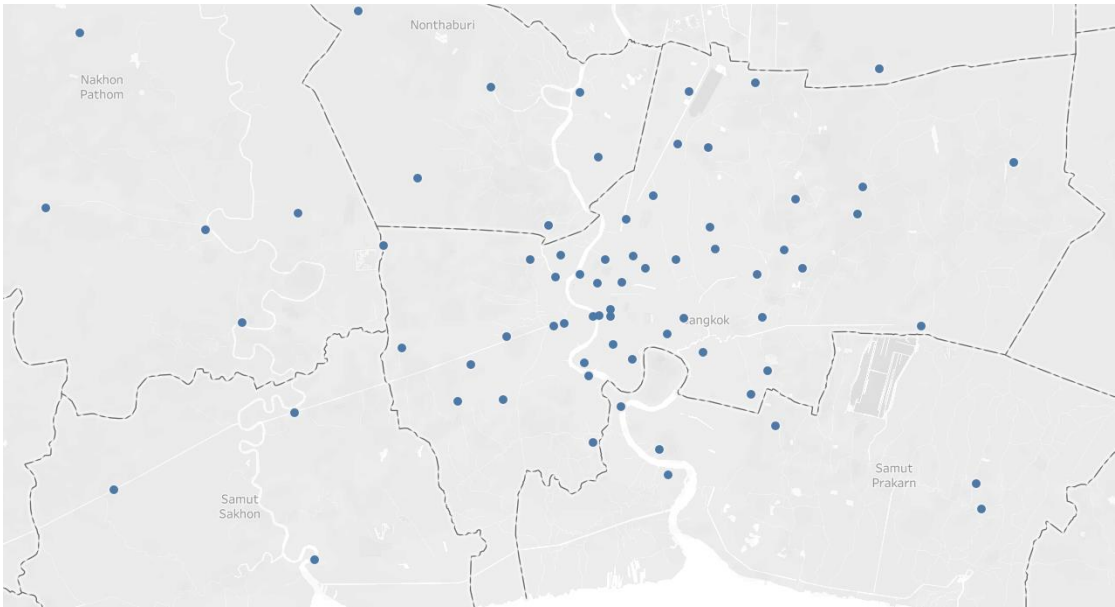
ภาพที่ 3.2: สัญญาณของจำนวนผู้ติดเชื้อรายสัปดาห์ในแต่ละอำเภอ

### 3.3 การสร้างกราฟ

นำข้อมูลตำแหน่งของสัญญาณหรือตำแหน่งของอำเภอมาสร้างเป็นกราฟ ภายใต้สมมติฐานที่ว่าอำเภอที่อยู่ใกล้กันจะมีจำนวนผู้ติดเชื้อใกล้เคียงกัน จุดยอดของกราฟคือตำแหน่งของอำเภอ และน้ำหนักเส้นคือระยะห่างระหว่างอำเภอ โดยอำเภอที่มีระยะห่างกันน้อยจะมีค่าน้ำหนักมาก และอำเภอที่มีระยะห่างกันมากจะมีน้ำหนักน้อย แต่เนื่องจากตำแหน่งของสัญญาณแทนตำแหน่งของอำเภอ ซึ่งอำเภอของประเทศไทยมีความหนาแน่นไม่เท่ากันในแต่ละพื้นที่ ดังภาพที่ 3.3 จะเห็นได้ว่าในบริเวณกรุงเทพมหานครมีจำนวนเขตมากกว่าจำนวนอำเภอในจังหวัดรอบข้างมาก จึงไม่สามารถใช้



สมการ (1) ในการกำหนดน้ำหนักสำหรับระยะห่างระหว่างอำเภอของประเทศไทยได้ เพราะน้ำหนักเส้นที่ได้ในแต่ละอำเภอจะแตกต่างกันตามความหนาแน่นของอำเภอในพื้นที่นั้นๆ เช่น น้ำหนักเส้นของเขตในกรุงเทพจะมีค่ามากที่สุดที่ 1 แต่น้ำหนักเส้นของอำเภอในจังหวัดนครปฐมมีค่ามากที่สุดที่ 0.62 จึงได้เสนอการกำหนดน้ำหนักด้วยสมการ (5) ซึ่งเป็นการกำหนดน้ำหนักที่ไม่อิงกับความหนาแน่นของอำเภอในพื้นที่นั้นๆ ผลลัพธ์ที่ได้จากสมการ (5) จะทำให้น้ำหนักเส้นของแต่ละอำเภอมีค่ามากที่สุดเท่ากันคือ 1 และจะไม่นำอำเภอที่มีระยะห่างมากกว่า 4 เท่าของระยะห่างที่น้อยที่สุดของอำเภอนั้นมาพิจารณาเป็นเส้นเชื่อม แต่เนื่องจากการคำนวณน้ำหนักในสมการ (5) มีการใช้ค่าจากระยะห่างที่สั้นที่สุดในอำเภอนั้น  $M_n$  ทำให้การกำหนดน้ำหนักระหว่างอำเภอ A ไปอำเภอ B และอำเภอ B ไปอำเภอ A อาจมีค่าไม่เท่ากัน กราฟที่ได้จากสมการ (5) จึงเป็นกราฟมีทิศทาง แต่จากการศึกษาพบว่าผลลัพธ์ที่ได้จากกระบวนการประมวลผลสัญญาณยังคงให้ผลที่ถูกต้อง จะได้กราฟที่มีจุดยอด 928 จุด เป็นกราฟมีน้ำหนัก และมีทิศทาง หรือเมทริกซ์  $W \in \mathbb{R}_+^{928 \times 928}$



ภาพที่ 3.3: ตำแหน่งของอำเภอในบริเวณจังหวัดกรุงเทพมหานคร

$$W_{nm} = \begin{cases} 1 - \log_4 \left( \frac{r_{nm}}{M_n} \right) & , r_{nm} < 4M_n \\ 0 & , r_{nm} \geq 4M_n \end{cases} \quad (5)$$

โดยที่  $W_{nm}$  เป็นน้ำหนักของเส้นเชื่อมระหว่างจุดยอด  $n, m$

$r_{nm}$  เป็นระยะห่าง Euclid ระหว่างจุดยอด  $n, m$

$M_n$  เป็นระยะห่างที่น้อยที่สุดของจุดยอด  $n$

### 3.4 การปรับสัญญาณกราฟเรียบแบบอุดมคติ

นำสัญญาณที่ได้จากหัวข้อ 3.1 และกราฟจากหัวข้อ 3.2 มาผ่านกระบวนการในหัวข้อ 2.1.3 และ 2.1.5 ทำการสร้างเมทริกซ์ Laplacian  $\mathbf{L}$  และแยกส่วนประกอบ eigenvector ของเมทริกซ์ Laplacian  $\mathbf{L}$  ได้เมทริกซ์ eigenvectors  $\mathbf{V}$  และเมทริกซ์ eigenvalues  $\mathbf{\Lambda}$  ทำการสร้าง ideal low-pass filter จำนวน 6 filter จากการเลือก eigenvectors มาปริมาณที่ต่างกันคือ 5%, 10%, 20%, 30%, 40% และ 80% ของปริมาณ eigenvectors ทั้งหมด เพื่อนำมาพิจารณาเลือก filter ที่เหมาะสมกับผลลัพธ์ที่ต้องการ จากนั้นนำ filter มากรองสัญญาณจะได้สัญญาณที่ผ่านการกรองมา 6 สัญญาณ โดยสัญญาณที่ได้จากการกรองจะมีความเรียบหรือความแปรปรวนทั้งหมดแตกต่างกันตามปริมาณ eigenvectors ที่ถูกเลือก

### 3.5 การกำหนดค่า magnitude ประจำอำเภอสำหรับนำไประบุรูปแบบการระบาด

ในขั้นตอนนี้เป็นขั้นตอนที่แสดงถึงการประยุกต์นำวิธีการประมวลสัญญาณกราฟมาใช้ในการระบุรูปแบบการระบาดโดยอ้างอิงวิธีการจากการศึกษาของ Yang Li และ Gonzalo Mateos นำผลลัพธ์ที่ได้จากการกรองสัญญาณในกรณีที่กำหนดมาสร้างเป็นค่าประจำอำเภอสำหรับระบุการระบาด ซึ่งก็คือการคำนวณ magnitude ประจำอำเภอโดยใช้สมการ (6) และ (7) ซึ่งคำนวณมาจากสัญญาณที่ผ่านการกรองในช่วงความถี่ต่ำ และความถี่สูง ส่งผลให้ได้ผลลัพธ์ออกมาเป็น 2 ชุดที่ใช้ในการระบุการระบาด คือ เมทริกซ์  $\mathbf{M} \in \mathbb{R}^{928 \times 2}$  ดังตารางที่ 3.2 และช่วงค่ายังแตกต่างกันในแต่ละสัปดาห์หากนำมาพิจารณารูปแบบการติดเชื้อแบบรายสัปดาห์ดังตารางที่ 3.3 โดยช่วงค่าจะแตกต่างกันตามจำนวนผู้ติดเชื้อที่เกิดขึ้นในแต่ละสัปดาห์ ทำให้การสรุปรูปแบบการระบาดในบางอำเภอทำได้ไม่ชัดเจน เช่น อำเภอที่ทั้ง magnitude ความถี่ต่ำมีค่าสูง และ magnitude ความถี่สูงมีค่าสูง ดังตารางที่ 3.3 ในสัปดาห์ที่ 14 ของเขตดุสิตทั้ง magnitude ความถี่ต่ำ และ magnitude ความถี่สูงมีค่าสูงด้วยกันทั้งคู่ทำให้สรุปได้ยากกว่ามีการระบาดรูปแบบใด

$$mag\_l(i) = \frac{\sum_{k=1}^{k=39} |x_l(i,k)|}{39} \quad (6)$$

$$mag\_h(i) = \frac{\sum_{k=1}^{k=39} |x_h(i,k)|}{39} \quad (7)$$

โดยที่  $mag\_l(i)$  เป็นค่า magnitude ความถี่ต่ำของอำเภอ  $i$

$mag\_h(i)$  เป็นค่า magnitude ความถี่สูงของอำเภอ  $i$

$x_l(i, k)$  เป็นสัญญาณความถี่ต่ำของอำเภอ  $i$  ที่สัปดาห์  $k$

$x_h(i, k)$  เป็นสัญญาณความถี่สูงของอำเภอ  $i$  ที่สัปดาห์  $k$

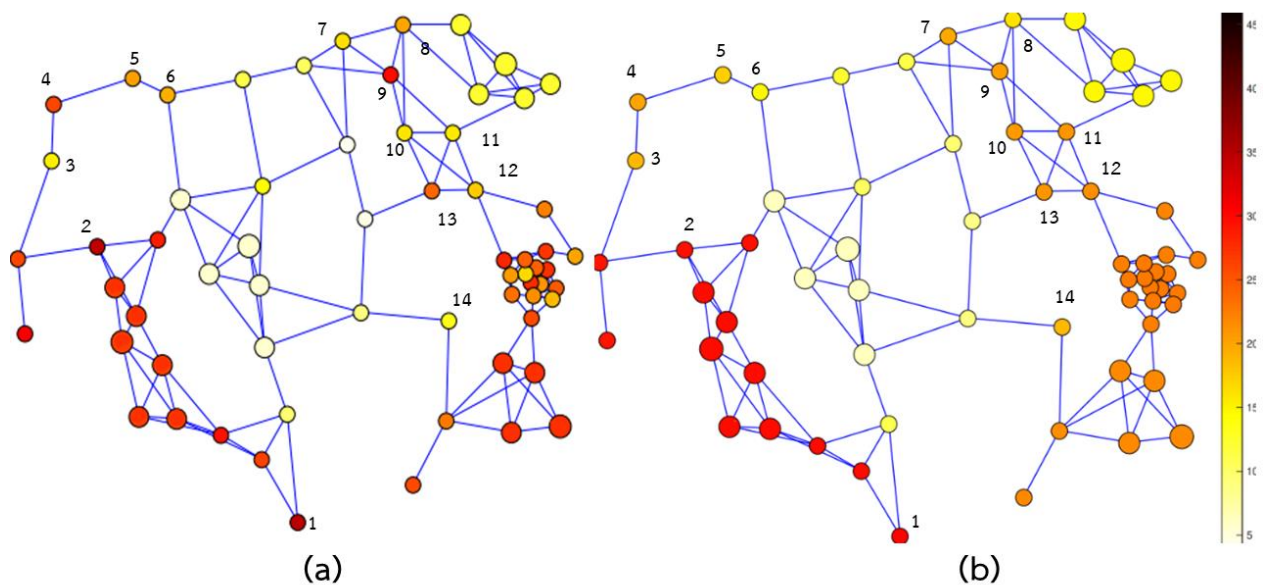
อำเภอ	Magnitude ความถี่ต่ำ	Magnitude ความถี่สูง
พระนครศรี	243.4912	54.79379
ดุสิต	241.4665	132.9184
หนองจอก	113.8908	16.46643
บางรัก	244.3803	130.0372
...	...	...
สุโขทัย	218.4308	10.91203
สุโขทัย	187.7202	1.830399
จะนะ	197.2508	5.419785
เกาะไอร้ง	273.8006	71.07828
range	[1.33,790]	[0.05,331]

ตารางที่ 3.2: ตัวอย่างค่า magnitude ทั้ง 39 สัปดาห์ ในแต่ละอำเภอ โดยมี  
อำเภอ ทั้งหมด 928 อำเภอ

อำเภอ	Magnitude ความถี่ต่ำ (week 13)	Magnitude ความถี่ต่ำ (week 14)	Magnitude ความถี่สูง (week 13)	Magnitude ความถี่สูง (week 14)
พระนครศรี	613.5928	656.3496	65.02625	31.6333
ดุสิต	708.4256	724.0485	295.2107	304.916
หนองจอก	165.795	337.051	45.63978	14.6242
บางรัก	600.5247	663.4243	531.7858	217.2946
บางเขน	119.405	184.6893	33.15971	108.4033
บางกะปิ	224.9806	223.94	76.52123	53.73461
...	...	...	...	...
ศรีสะเกษ	160.1832	251.6478	127.8336	193.5183
ฉะเชิงเทรา	105.1086	168.2391	5.8906	5.030337
สุรินทร์	104.4029	172.9662	9.958205	15.13551
สุโขทัย	105.5732	165.4534	5.671242	4.44712
สุโขทัย	105.1902	167.9675	8.783313	7.600185
จะนะ	106.5173	189.4295	22.32254	58.59846
เกาะไอร้ง	104.5881	192.9633	59.3351	4.89091
range	[0.02,832.36]	[0.002,1122.56]	[0,566.53]	[0.002,352.32]

ตารางที่ 3.3: ตัวอย่างค่า magnitude สัปดาห์ที่ 13 และ 14 ในแต่ละอำเภอโดย  
มีอำเภอทั้งหมด 928 อำเภอ

จากผลลัพธ์ที่ได้จากการปรับสัญญาณให้เรียบเปรียบเสมือนการทำ localized averaging เมื่อพิจารณาสัญญาณกราฟตั้งต้นเทียบกับสัญญาณกราฟที่ผ่านการกรอง จากภาพที่ 2.1 และภาพที่ 2.2 พบว่าจุดยอดที่สัญญาณต่างกับจุดยอดรอบข้างมาก ค่าสัญญาณที่ผ่านการกรองจะถูกปรับให้เปลี่ยนไปจากเดิมมาก ในทางตรงกันข้ามจุดยอดที่มีสัญญาณใกล้เคียงจุดยอดรอบข้าง ค่าที่ได้จากสัญญาณที่ผ่านการกรองจะใกล้เคียงกับค่าดั้งเดิม ภาพที่ 3.4 แสดงตัวอย่างจุดยอดที่ถูกปรับให้ค่าใกล้เคียงกับค่ารอบข้างส่งผลให้ค่าเปลี่ยนไปจากเดิม ซึ่งหากสร้าง filter ด้วยความถี่ต่ำ ๆ หรือสร้างจากการเลือกให้ eigenvectors มีปริมาณน้อยลง ผลลัพธ์ที่ได้จะทำให้พบจุดที่ต่างไปจากเดิมมากขึ้นจากการที่สัญญาณถูกปรับให้ค่าใกล้เคียงกันมากขึ้น



ภาพที่ 3.4: จุดยอดที่ถูกปรับให้มีค่าใกล้เคียงกับค่าในบริเวณรอบข้าง โดยภาพ (a) แสดงสัญญาณกราฟตั้งต้น และภาพ(b) แสดงสัญญาณกราฟที่ผ่านการกรอง

อาศัยหลักการความต่างระหว่างค่าดั้งเดิมกับค่าที่ผ่านการกรองนี้ในการกำหนดค่า magnitude เพื่อแสดงถึงรูปแบบการกระโดดโดยใช้สัญญาณในช่วงความถี่ต่ำเพียงช่วงเดียว ผลที่ได้คือ สมการ (8) สำหรับใช้ในการกำหนดค่า magnitude ในแต่ละอำเภอ ซึ่งคำนวณมาจาก  $\log_2$  ของ อัตราส่วนระหว่างผลรวมของค่าสมบูรณ์ของจำนวนผู้ติดเชื้อที่ผ่านการปรับให้เรียบ กับ ผลรวมจำนวนผู้ติดเชื้อดั้งเดิม แทนการใช้ค่าเฉลี่ยของค่าสมบูรณ์ของจำนวนผู้ติดเชื้อที่ผ่านการกรองดังสมการ (6) ได้ผลลัพธ์คือ  $\mathbf{M} \in \mathbb{R}^{928 \times 1}$  โดยผลลัพธ์ที่ได้จากสมการ (8) จะมีช่วงค่าเดียวที่สามารถอธิบายถึงรูปแบบการกระโดดที่แตกต่างกันคือ  $[-1, 1]$  ดังตารางที่ 3.4 โดยหากค่าใกล้เคียงกับ 0 หมายความว่า จำนวนผู้ติดเชื้อภายในอำเภอมีค่าใกล้เคียงกับอำเภอรอบข้าง และอำเภอที่มีช่วงค่าเข้าใกล้ -1 หรือ 1 จะแสดงถึงอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างกับอำเภอรอบข้าง โดยหากมีค่าเข้าใกล้ -1 หมายความว่า

ว่ามีจำนวนผู้ติดเชื้อภายในอำเภอน้อยกว่าอำเภอรอบข้าง และค่าเข้าใกล้ 1 หมายความว่า มีจำนวนผู้ติดเชื้อภายในอำเภอมากกว่าอำเภอรอบข้าง ซึ่งการกำหนด magnitude ดังสมการ (8) ทำให้สามารถแยกความแตกต่างของอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างกันได้ละเอียดมากขึ้น ส่งผลให้สามารถระบุการระบาดได้ 3 รูปแบบ คือ อำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับอำเภอรอบข้าง อำเภอที่มีจำนวนผู้ติดเชื่อน้อยกว่าอำเภอรอบข้าง และอำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าอำเภอรอบข้าง ซึ่งมากกว่าผลลัพธ์ที่ได้จากขั้นตอนวิธีเดิมซึ่งระบุได้เพียง 2 รูปแบบ คือ อำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับอำเภอรอบข้าง และอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างจากอำเภอรอบข้าง

$$mag(i) = \log_2 \left( \frac{(\sum_{k=1}^{k=39} x(i,k)) + p}{(\sum_{k=1}^{k=39} |x_t(i,k)|) + p} \right) \quad (8)$$

โดยที่  $mag(i)$  เป็นค่า magnitude ของอำเภอที่  $i$  โดยค่าที่น้อยกว่า -1 จะถูกปรับให้เป็น -1 และค่าที่มากกว่า 1 จะถูกปรับให้เป็น 1

อำเภอ	magnitude
พระนครศรี	0.036178
ดุสิต	-0.41774
หนองจอก	0.018639
บางรัก	-0.19083
บางเขน	-0.76348
บางกะปิ	0.303265
...	...
แว้ง	-0.34222
สุคริณ	0.165153
สุโขทัย	-0.12174
สุโขทัย	-0.29562
จะนะ	0.547842
เจาะไอร้อง	-0.24139
range	[-1,1]

ตารางที่ 3.4: ตัวอย่างค่า magnitude ในแต่ละอำเภอโดยมีอำเภอทั้งหมด 928 อำเภอ ที่คำนวณได้จากสมการ (8)

ผลลัพธ์ที่ได้จากสมการ (8) จะเป็นการระบุรูปแบบการระบาดในแต่ละอำเภอแบบพิจารณา รวมทั้ง 39 สัปดาห์ซึ่งได้ผลลัพธ์ออกมาเป็นค่าเดียวในแต่ละอำเภอ หากต้องการระบุรูปแบบการระบาดรายสัปดาห์จะสามารถหาได้จากสมการ (9) โดยเปลี่ยนจากการหาผลรวมเป็นการพิจารณาทีละสัปดาห์หรือทีละสมาชิกแทน จะได้ผลลัพธ์ออกมาเป็นเมทริกซ์ magnitude  $\mathbf{M} \in \mathbb{R}^{928 \times 39}$  โดยผลลัพธ์ที่ได้ในแต่ละสัปดาห์ยังคงตั้งอยู่ในช่วงค่าเดียวคือ  $[-1, 1]$  ดังตารางที่ 3.5 ซึ่งแตกต่างจากวิธีที่งานของ Yang Li และ Gonzalo Mateos โดยคำนวณจากสมการ (6) และ (7) แต่แบ่งคำนวณสร้างทีละสัปดาห์แทน ซึ่งจะได้ผลลัพธ์ในแต่ละสัปดาห์แตกต่างกันตามจำนวนผู้ติดเชื้อที่เกิดขึ้นในแต่ละสัปดาห์ดังตารางที่ 3.3

$$mag(i, j) = \log_2 \left( \frac{x(i, k) + q}{|x_l(i, k)| + q} \right) \quad (9)$$

โดยที่  $mag(i, j)$  เป็นค่า magnitude ของอำเภอที่  $i$  และสัปดาห์ที่  $k$   
โดยค่าที่น้อย -1 จะถูกปรับให้เป็น -1 และค่าที่มากกว่า 1 จะถูกปรับให้เป็น 1

อำเภอ	magnitude (week 1)	magnitude (week 2)	...	magnitude (week 38)	magnitude (week 39)
พระนคร	-0.33	-0.40743		0.310642	0.161793
ดุสิต	0.230631	-0.688		-0.30764	-0.87911
หนองจอก	-0.10312	-0.24417		-0.88062	-0.28009
บางรัก	0.052644	-0.19411		0.204941	0.278488
บางเขน	0.046361	0.358751		-0.10891	0.076209
บางกะปิ	-0.11831	-0.20707		-0.26879	-0.04809
ปทุมวัน	0.135025	0.262085		0.449481	1
...	...	...	...	...	...
ศรีสาคร	-0.50335	-0.67824		-0.85526	-0.2973
แว้ง	-0.19261	-0.13296		-0.0374	0.021745
สุคิริน	-0.25683	-0.32014		-0.50586	-0.38469
สุโขทัย	-0.0939	-0.03047		0.283495	0.338987
สุโขทัย	-0.19149	-0.13159		-0.14774	-0.03655
จะนะ	-0.53311	-0.70851		0.857747	-0.05922
เจาะไอร้อง	-0.63766	-0.84723		-0.92159	-0.42623
range	$[-1, 1]$	$[-1, 1]$	$[-1, 1]$	$[-1, 1]$	$[-1, 1]$

ตารางที่ 3.5: ตัวอย่างค่า magnitude รายสัปดาห์ 39 สัปดาห์ในแต่ละอำเภอ  
โดยมีอำเภอทั้งหมด 928 อำเภอ ที่คำนวณได้จากสมการ (9)

สมการ (8) และ (9) มีการบวกด้วยค่าคงตัว  $p$  และ  $q$  ในพจน์ด้านบนและด้านล่างเพื่อแก้ไขในกรณีที่จำนวนผู้ติดเชื่อ หรือจำนวนผู้ติดเชื่อสะสมมีค่าต่ำมาก จะทำให้อัตราส่วนความต่างระหว่างสัญญาณดั้งเดิมและสัญญาณที่ผ่านการกรอง มีค่าต่ำมาก หรือสูงมากตามความต่างที่เกิดขึ้น เนื่องจากผลลัพธ์ที่ได้จากสัญญาณที่ผ่านการกรองเปรียบเสมือนกับการทำ localized averaging ซึ่งเป็นการปรับค่าแบบเชิงเส้น แต่สมการ (8) และ (9) เป็นการเปรียบเทียบแบบอัตราส่วน ส่งผลให้ magnitude ที่ได้จะ bias กับอำเภอที่มีจำนวนผู้ติดเชื่อน้อยๆ การบวกด้วยค่าคงตัวจะช่วยปรับค่าให้การกำหนด magnitude ระหว่างอำเภอที่มีจำนวนผู้ติดเชื่อน้อยๆ และจำนวนผู้ติดเชื่อมากๆ มีความเท่าเทียมกันมากขึ้น โดยในที่นี้จะกำหนดให้  $p = 500$  และ  $q = 30$

## บทที่ 4

### ผลการวิจัย

ในบทนี้จะกล่าวถึง ผลการทดลองที่ได้จากการปรับสัญญาณเรียบด้วย filter ที่แตกต่างกัน และการทดสอบสมมติฐาน T-test ของความแปรปรวนประจำอำเภอ และจำนวนผู้ติดเชื้อเฉลี่ยภายในอำเภอ ระหว่างรูปแบบการระบาดที่ระบุได้ 3 รูปแบบ จากนั้นนำค่า magnitude ที่ได้ไปสร้าง heat map สำหรับนำไปแปลผลเพื่อระบุการระบาดในแต่ละอำเภอ

#### 4.1 ผลการทดลองการปรับสัญญาณเรียบด้วย filter ที่แตกต่างกัน

สัญญาณที่กรองจาก filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณน้อย จะมีความเรียบมากกว่า filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณมาก ดังนั้นในการทดลองนี้ สัญญาณที่กรองด้วย filter ที่สร้างจากการเลือกปริมาณ eigenvectors มา 5% จากปริมาณ eigenvectors ทั้งหมด จะมีความเรียบมากที่สุด กล่าวคือความแปรปรวนทั้งหมดมีค่าน้อยลงที่สุดเมื่อเทียบกับความแปรปรวนทั้งหมดของสัญญาณกราฟดั้งเดิม ดังตารางที่ 4.1 และได้แสดงภาพผลลัพธ์สัญญาณที่ถูกปรับให้เรียบที่ได้ในแต่ละ filter ในภาคผนวก ข

ตารางที่ 4.1: ความแปรปรวนทั้งหมดของสัญญาณที่ผ่านการกรองด้วย filter ที่ต่างกันต่อความแปรปรวนทั้งหมดของสัญญาณตั้งต้น

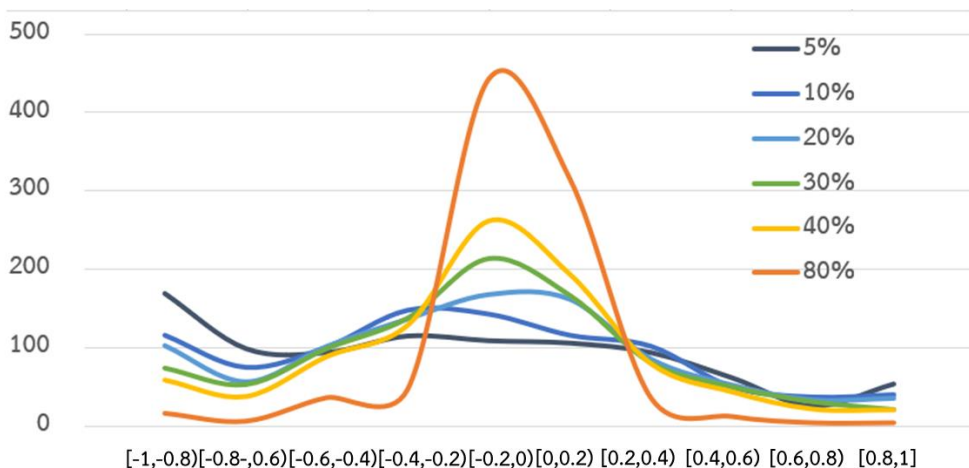
ปริมาณ eigenvectors	ความแปรปรวนทั้งหมดของสัญญาณที่ผ่านการกรองต่อความแปรปรวนทั้งหมดของสัญญาณตั้งต้น
5%	0.021
10%	0.055
20%	0.211
30%	0.301
40%	0.546
80%	0.838



ความเรียบของสัญญาณจะส่งผลต่อการกระจายค่าของ magnitude เนื่องจาก magnitude ที่คำนวณจากสมการ (8) และ (9) มีการใช้ความแตกต่างระหว่างสัญญาณตั้งต้น และสัญญาณที่ถูกปรับให้เรียบ ซึ่งสัญญาณที่ถูกปรับให้เรียบมากหรือสัญญาณที่กรองจาก filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณน้อย ค่าสัญญาณจะถูกปรับให้ต่างไปจากเดิมมากกว่าสัญญาณที่ถูกกรองด้วย filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณมาก ส่งผลให้สัญญาณที่กรองจาก filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณน้อย จะมีความถี่ของค่า magnitude ในช่วงขอบของค่าหรือ  $[-1, 0.8]$  มากกว่าสัญญาณที่กรองจาก filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณมาก พิจารณาได้ว่า ผลลัพธ์ที่ได้จาก filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณน้อย จะมีจำนวนอำเภอที่จำนวนผู้ติดเชื้อแตกต่างจากรอบข้างมากกว่าผลลัพธ์ที่ได้จาก filter ที่สร้างจากการเลือก eigenvectors ให้มีปริมาณมาก ดังตารางที่ 4.2 และภาพที่ 4.1 จะพบว่าในช่วงค่า  $[-1, -0.8]$  และ  $(0.8, 1]$  ของ filters ที่สร้างขึ้นจากการเลือก eigenvectors ให้มีปริมาณน้อย จะมีความถี่มากกว่าหรือมีจำนวนอำเภอมากกว่า filters ที่สร้างขึ้นจากการเลือก eigenvectors ให้มีปริมาณมาก

**ตารางที่ 4.2: การแจกแจงความถี่ของค่า magnitude ที่สร้างได้จาก filter ที่ต่างกัน**

ช่วงค่า	ความถี่ที่เกิดขึ้นในแต่ละ filters					
	0.05	0.1	0.2	0.3	0.4	0.8
$[-1, -0.8]$	169	116	102	74	58	17
$[-0.8, -0.6]$	99	75	56	53	37	7
$[-0.6, -0.4]$	94	101	101	99	88	37
$[-0.4, -0.2]$	115	148	136	138	128	48
$[-0.2, 0]$	109	143	167	214	261	443
$[0, 0.2]$	106	116	161	167	193	316
$[0.2, 0.4]$	94	102	85	81	79	37
$[0.4, 0.6]$	62	50	52	50	43	13
$[0.6, 0.8]$	26	37	33	31	21	5
$[0.8, 1]$	54	40	35	21	20	5



ภาพที่ 4.1: การแจกแจงความถี่ของค่า magnitude ของสัญญาณที่กรองจาก filter ที่กำหนด

ข้อมูลที่นำมาใช้ในการทดลองนี้ เลือกมาจากข้อมูลในช่วงเริ่มการระบาดระลอกที่ 3 ซึ่งเป็นการระบาดที่เชื้อไวรัสกลายพันธุ์กลายเป็นสายพันธุ์ delta ซึ่งมีความสามารถในการแพร่กระจายที่รวดเร็วมาก ส่งผลให้อำเภอที่มีจำนวนผู้ติดเชื้อจะมีความใกล้เคียงกันมีอยู่เป็นจำนวนมาก และอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างจากอำเภอที่ใกล้เคียงกันมีอยู่เป็นจำนวนน้อย ทั้งนี้อำเภอแต่ละอำเภอมีความสามารถในการควบคุมการระบาดแตกต่างกัน จึงทำให้พบอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างจากอำเภอรอบข้างอยู่เป็นจำนวนหนึ่ง ดังนั้นผลลัพธ์ที่เหมาะสมสำหรับนำมาระบุการระบาดคือผลลัพธ์ที่มีจำนวนอำเภอที่จำนวนผู้ติดเชื้อใกล้เคียงกันมากกว่าจำนวนอำเภอที่จำนวนผู้ติดเชื้อแตกต่างจากรอบข้าง หรือความถี่ในช่วง  $[-1,0.8]$  และ  $(0.8,1]$  มีค่าน้อยกว่าความถี่ในช่วง  $[-0.2,0.2]$  และต้องมีจำนวนอำเภอที่จำนวนผู้ติดเชื้อแตกต่างกับอำเภอรอบข้างไม่น้อยจนเกินไป หรือความถี่ในช่วง  $[-1,0.8]$  และ  $(0.8,1]$  มีค่าไม่น้อยจนเกินไป

จากการพิจารณาการแจกแจงความถี่ของ magnitude ที่สร้างจาก filters ต่างๆ และลักษณะการระบาดในประเทศไทย ทำให้สรุปได้ว่าการกระจายค่าของ magnitude ที่สร้างด้วย filters จากการเลือกปริมาณ eigenvectors มา 30% ให้ผลลัพธ์ที่เหมาะสมกับลักษณะการระบาดของข้อมูลที่นำมาสร้างเป็นสัญญาณมากที่สุด

## 4.2 การทดสอบสมมติฐาน T-test

ขั้นตอนนี้เป็นขั้นตอนทดสอบเพื่อยืนยันผลลัพธ์ที่ได้จากการกำหนด magnitude ด้วยสมการ (8) และสร้างจาก filter ที่เลือกปริมาณ eigenvectors มา 30% สามารถนำมาใช้ในการระบุรูปแบบการระบาดได้ โดยจะกำหนดสมมติฐานความสามารถในการระบุรูปแบบการระบาด 3 รูปแบบในแต่ละช่วงค่าดังนี้

ค่าใกล้เคียง 0 จะแสดงถึงอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับบริเวณรอบข้าง

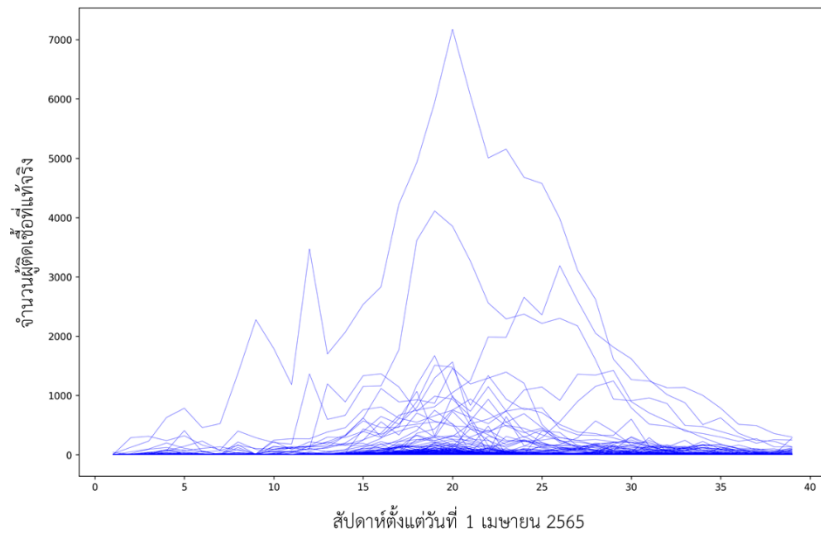
ค่าในช่วง -1 และ 1 จะแสดงถึงอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างจากบริเวณรอบข้าง โดยหากมีค่าใกล้เคียง -1 คืออำเภอที่มีจำนวนผู้ติดเชื่อน้อยกว่าบริเวณรอบข้าง และหากมีค่าใกล้เคียง 1 คืออำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง

ทำการทดสอบโดยเลือกกลุ่มข้อมูลมา 3 กลุ่ม กลุ่มละ 100 อำเภอ โดยกลุ่มข้อมูลที่ 1 คือกลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกันในบริเวณรอบข้าง สร้างมาจากการเลือกอำเภอที่มีค่า magnitude ใกล้เคียง 0 ที่สุดมา 100 อำเภอ กลุ่มข้อมูลที่ 2 คือกลุ่มข้อมูลที่แสดงอำเภอที่มีจำนวนผู้ติดเชื่อน้อยกว่าบริเวณรอบข้าง สร้างมาจากการเลือกอำเภอที่มีค่า magnitude น้อยที่สุด 100 อำเภอ และกลุ่มข้อมูลที่ 3 แทนกลุ่มข้อมูลของอำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง สร้างมาจากการเลือกอำเภอที่มีค่า magnitude มากที่สุด 100 อำเภอ จะได้ตัวอย่างกลุ่มข้อมูลดังตารางที่ 4.3

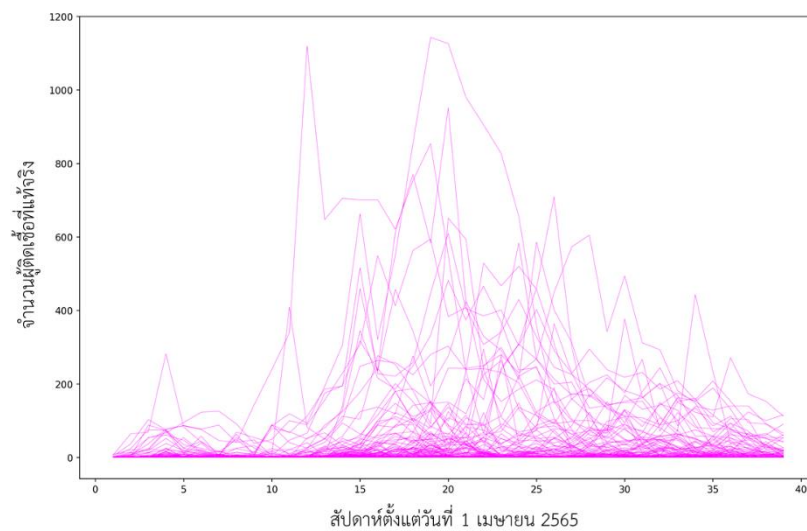
อำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับบริเวณรอบข้าง	อำเภอที่มีจำนวนผู้ติดเชื่อน้อยกว่าบริเวณรอบข้าง	อำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง
พระนครศรี(0.036)	บางเขน(-0.763)	ปทุมวัน(0.901)
หนองจอก(0.018)	ตลิ่งชัน(-0.839)	ป้อมปราบศัตรูพ่าย(0.602)
ดินแดง(-0.048)	หนองแขม(-0.894)	มีนบุรี(0.498)
จอมทอง(0.022)	สะพาน(-1.0)	เมืองนครศรีธรรมราช(0.446)
คลองสามวา(0.021)	แก่งหางแมว(-1.0)	พยุหะภูมิพิสัย(0.420)
...	...	...
ทองผาภูมิ(-0.036)	ห้างฉัตร(-0.887)	เมืองจันทบุรี(1.0)
สังขละบุรี(0.010)	สะบ้าย้อย(-0.689)	วารินชำราบ(1.0)
ดอนตูม(0.010)	เมืองพัทลุง(-1.0)	บันนังสตา(0.668)
เมืองสมุทรสงคราม(-0.041)	ศรีนครินทร์(-1.0)	ธารโต(0.447)
ชะอำ(-0.009)	รือเสาะ(-1.0)	สุโขทัย(0.547)

ตารางที่ 4.3: ตัวอย่างกลุ่มข้อมูลที่นำมาทดสอบค่าที่แสดงในวงเล็บคือค่า magnitude ประจำอำเภอ

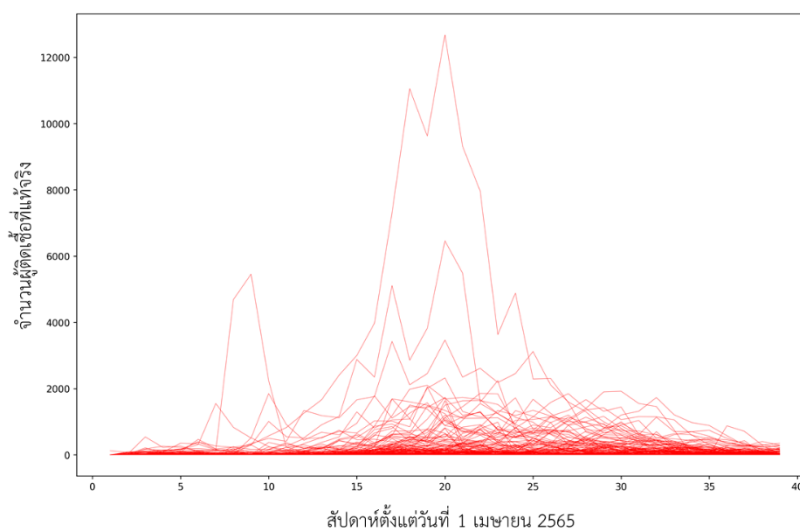
โดยชุดข้อมูลที่น่ามาทำการทดสอบคือข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงที่ไม่ได้ผ่านการทำ normalize เพื่อเป็นการทดสอบยืนยันว่าผลลัพธ์ที่ได้จากวิธีการที่เสนอสามารถนำไปประนูปแบบการระบาดของข้อมูลที่แท้จริงได้ และทำการทดสอบ T-test ด้วย  $p\text{-value} = 0.05$  ได้ภาพสัญญาณในแต่ละกลุ่มข้อมูล ดังภาพที่ 4.2 - 4.4



ภาพที่ 4.2: สัญญาณที่สร้างจากข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงของกลุ่มข้อมูลที่ 1 จะเห็นได้ว่าสัญญาณมีการเพิ่มและลดของจำนวนผู้ติดเชื้อในลักษณะเดียวกัน



ภาพที่ 4.3: สัญญาณที่สร้างจากข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงของกลุ่มข้อมูลที่ 2 จะเห็นได้ว่าลักษณะสัญญาณไม่มีความเกี่ยวเนื่องกัน ทำให้เห็น spike ของสัญญาณ และมีจำนวนผู้ติดเชื่อน้อยกว่ากลุ่มข้อมูลที่ 1



ภาพที่ 4.4: สัญญาณที่สร้างจากข้อมูลจำนวนผู้ติดเชื้อที่แท้จริงของกลุ่มข้อมูลที่ 3 จะเห็นได้ว่าสัญญาณมีการเพิ่มและลดของจำนวนผู้ติดเชื้อในลักษณะเดียวกัน แต่มีจำนวนผู้ติดเชื้อมากกว่ากลุ่มข้อมูลที่ 1

#### 4.2.1 การทดสอบความต่างของความแปรปรวนประจำอำเภอระหว่างการติดเชื้อรูปแบบทั้ง 3 รูปแบบ

การทดสอบนี้เป็นการทดสอบเพื่อยืนยันว่ากลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกันในบริเวณรอบข้าง หรือกลุ่มข้อมูลที่ 1 มีความแปรปรวนประจำอำเภอน้อยกว่า กลุ่มข้อมูลที่ 2 และ 3 หรือกลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างจากบริเวณรอบข้าง กล่าวคือเป็นการทดสอบเพื่อยืนยันว่า อำเภอที่มีค่า magnitude ใกล้เคียง 0 จะมีความแตกต่างของจำนวนผู้ติดเชื้อในบริเวณใกล้เคียงกันน้อยกว่าอำเภอที่มีค่าใกล้เคียง -1 และ 1 หรือเป็นการทดสอบยืนยันว่าอำเภอที่มีค่าใกล้เคียง 0 จะสามารถแทนถึงอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับบริเวณรอบข้างได้ โดยคำนวณความแปรปรวนประจำอำเภอจากสมการ (10) ซึ่งดัดแปลงมาจากการคำนวณความแปรปรวนทั้งหมด สมการ (2) โดยได้เปลี่ยนการพิจารณาความต่างระหว่างค่าด้วยผลลบเป็นอัตราส่วนแทน ค่าที่ได้จะแสดงถึงปริมาณความต่างของจำนวนผู้ติดเชื้อประจำอำเภอเมื่อเทียบกับบริเวณที่ใกล้เคียงกัน จะได้ตัวอย่างของค่าความแปรปรวนประจำอำเภอในแต่ละกลุ่มข้อมูลดังตารางที่ 4.4

$$TV\_r(i) = \sum_{k=1}^{39} \sum_{j=1}^{928} w(i,j) \times |\log_2 \left( \frac{x(i,k)+1}{x(j,k)+1} \right)| \quad (10)$$

โดยที่  $TV\_r(i)$  เป็นความแปรปรวนประจำอำเภอ  $i$

$w(i,j)$  เป็นน้ำหนักระหว่างอำเภอ  $i$  และอำเภอ  $j$

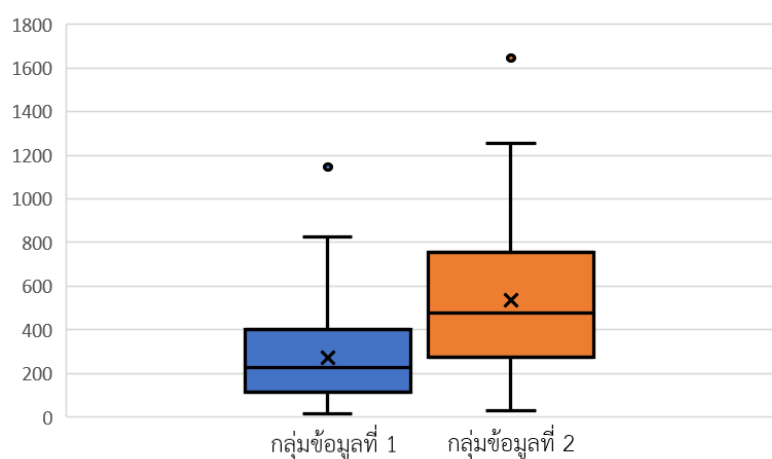
$x(i,k)$  เป็นจำนวนผู้ติดเชื้ออำเภอ  $i$  ที่สัปดาห์  $k$

$x(j,k)$  เป็นจำนวนผู้ติดเชื้ออำเภอ  $j$  ที่สัปดาห์  $k$

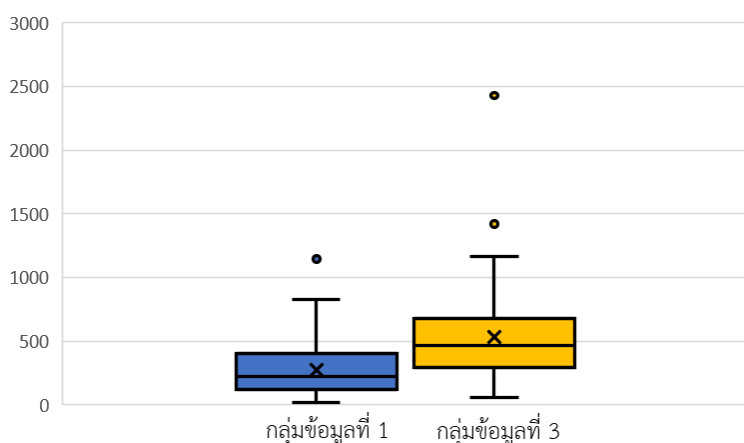
อำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับบริเวณรอบข้าง	อำเภอที่มีจำนวนผู้ติดเชื้อน้อยกว่าบริเวณรอบข้าง	อำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง
พระนครศรี(368)	บางเขน(1247)	ปทุมวัน(479)
หนองจอก(285)	ตลิ่งชัน(1253)	ป้อมปราบศัตรูพ่าย(461)
ดินแดง(83)	หนองแขม(1153)	มีนบุรี(305)
จอมทอง(603)	สะพาน(296)	เมืองนครศรีธรรมราช(810)
คลองสามวา(229)	แก่งหางแมว(942)	พยุหะภูมิพิสัย(809)
...	...	...
ทองผาภูมิ(292)	ห้างฉัตร(388)	เมืองจันทบุรี(526)
สังขละบุรี(369)	สวายไวย(291)	วารินชำราบ(344)
ดอนตูม(138)	เมืองพัทลุง(765)	บันนังสตา(616)
เมืองสมุทรสงคราม(398)	ศรีนครินทร์(850)	ธารโต(351)
ชะอำ(126)	รือเสาะ(607)	สุโขทัย(1434)

ตารางที่ 4.4: ตัวอย่างค่าความแปรปรวนประจำอำเภอในแต่ละกลุ่มข้อมูลที่นำมาทดสอบค่าที่แสดงในวงเล็บคือค่าความแปรปรวนประจำอำเภอ

ทำการทดสอบ T-test ทั้งหมด 2 ครั้งโดยการทดสอบครั้งแรกคือการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 ได้ผลลัพธ์ดังภาพที่ 4.5 ครั้งที่สองคือการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 ได้ผลลัพธ์ดังภาพที่ 4.6



ภาพที่ 4.5: การกระจายค่าความแปรปรวนประจำอำเภอของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 โดยผ่านการทดสอบ T-test ที่  $p \text{ value} = 0.05$  จะเห็นได้ว่ากลุ่มข้อมูลที่ 1 มีความแปรปรวนประจำอำเภอน้อยกว่ากลุ่มข้อมูลที่ 2



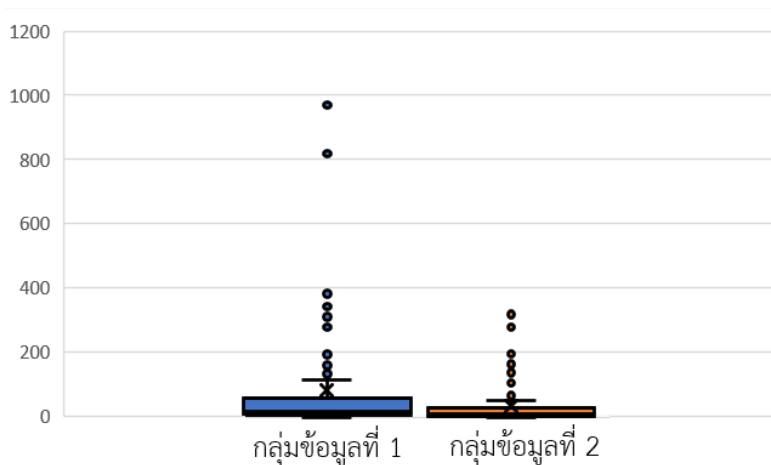
ภาพที่ 4.6: การกระจายค่าความแปรปรวนประจำอำเภอของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 โดยผ่านการทดสอบ T-test ที่  $p \text{ value} = 0.05$  จะเห็นได้ว่ากลุ่มข้อมูลที่ 1 มีความแปรปรวนประจำอำเภอน้อยกว่ากลุ่มข้อมูลที่ 3

จากการทดสอบแสดงให้เห็นว่ากลุ่มข้อมูลที่ 1 มีความแปรปรวนประจำอำเภอต่ำกว่ากลุ่มข้อมูลที่ 2 และกลุ่มข้อมูลที่ 3 ทำให้สามารถสรุปได้ว่ากลุ่มข้อมูลที่ 1 หรืออำเภอที่มีค่า magnitude ใกล้เคียง 0 แสดงถึงอำเภอที่มีจำนวนผู้ติดใกล้เคียงกับบริเวณรอบข้าง

#### 4.2.2 การทดสอบความต่างของจำนวนผู้ติดเชื้อเฉลี่ยประจำอำเภอระหว่างการติดเชื้อรูปแบบทั้ง 3 รูปแบบ

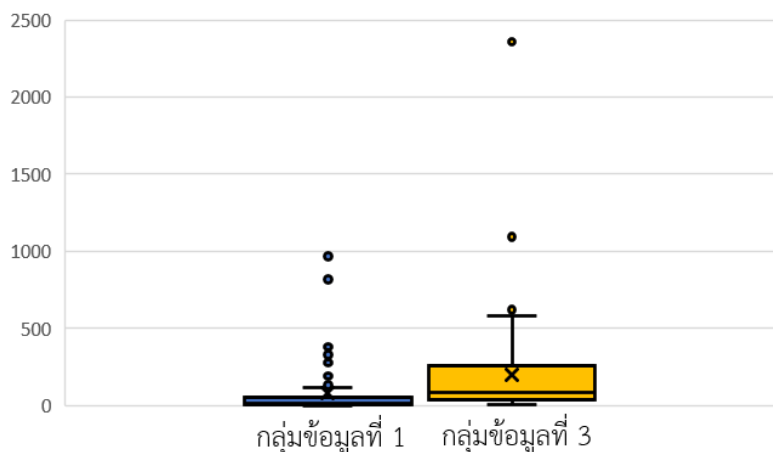
การทดสอบนี้เป็นการทดสอบเพื่อยืนยันว่ากลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อน้อยกว่าบริเวณรอบข้าง มีจำนวนผู้ติดเชื้อเฉลี่ยน้อยกว่ากลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกันในบริเวณรอบข้าง และกลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง มีจำนวนผู้ติดเชื้อเฉลี่ยมากกว่ากลุ่มข้อมูลที่แทนอำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกันในบริเวณรอบข้าง กล่าวคือหากให้จำนวนผู้ติดเชื้อเฉลี่ยของกลุ่มข้อมูลที่ 1 2 และ 3 คือ  $n_1$   $n_2$  และ  $n_3$  ตามลำดับ จะได้ว่า  $n_2 < n_1 < n_3$  เพื่อที่จะสรุปว่า อำเภอที่มีค่า magnitude ใกล้เคียง -1 แทนอำเภอที่มีจำนวนผู้ติดเชื้อน้อยกว่าบริเวณรอบข้าง และอำเภอที่มีค่า magnitude ใกล้เคียง 1 แทนอำเภอที่มีจำนวนผู้ติดเชื้อเฉลี่ยมากกว่าบริเวณรอบข้าง

ทำการทดสอบ T-test ทั้งหมด 2 ครั้งโดยการทดสอบครั้งแรกคือการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 ได้ผลลัพธ์ดังภาพที่ 4.7 ครั้งที่สองคือการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 ได้ผลลัพธ์ดังภาพที่ 4.8



ภาพที่ 4.7: การกระจายจำนวนผู้ติดเชื้อเฉลี่ยของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 โดยผ่านการทดสอบ T-test ที่  $p \text{ value} = 0.05$  จะเห็นได้ว่ากลุ่มข้อมูลที่ 2 มีจำนวนผู้ติดเชื้อเฉลี่ยน้อยกว่ากลุ่มข้อมูลที่ 1



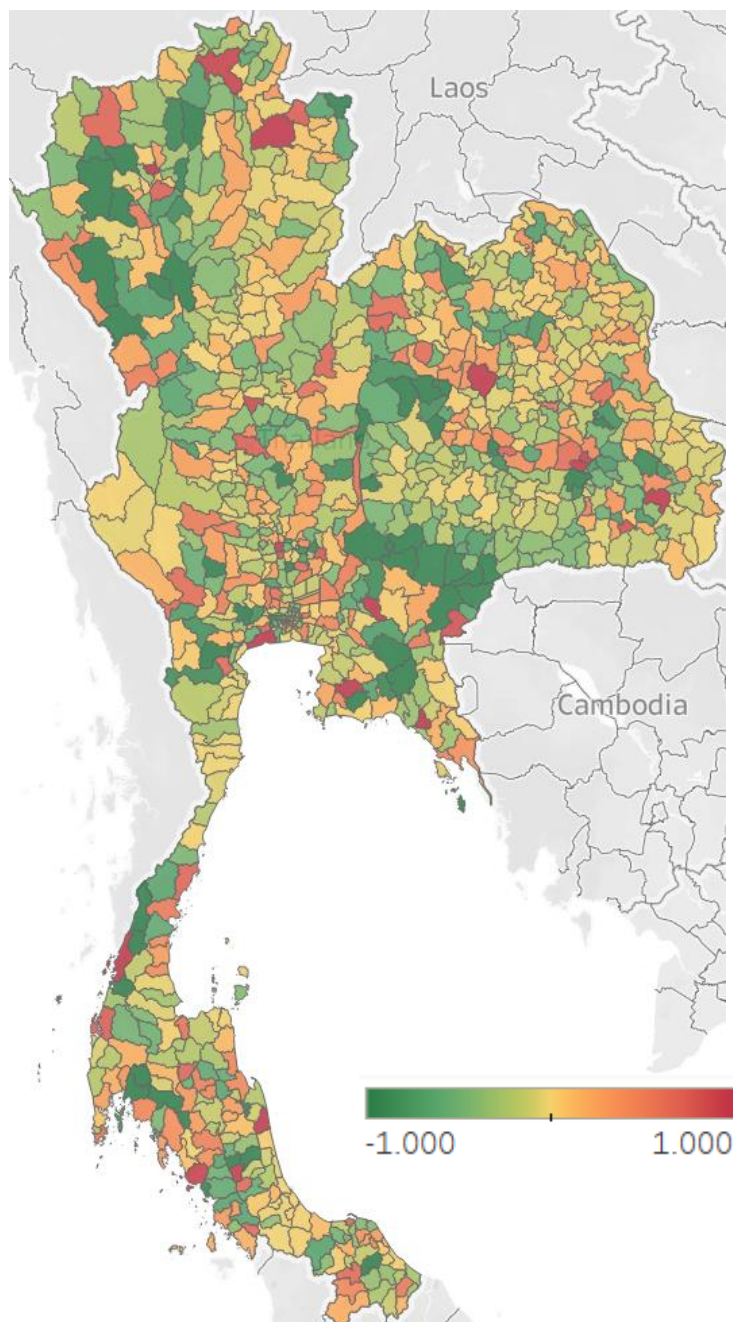


ภาพที่ 4.8: การกระจายจำนวนผู้ติดเชื้อเฉลี่ยของกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูล ที่ 3 โดยผ่านการทดสอบ T-test ที่  $p \text{ value} = 0.05$  จะเห็นได้ว่ากลุ่มข้อมูลที่ 1 มีจำนวนผู้ติดเชื้อเฉลี่ยน้อยกว่ากลุ่มข้อมูลที่ 3

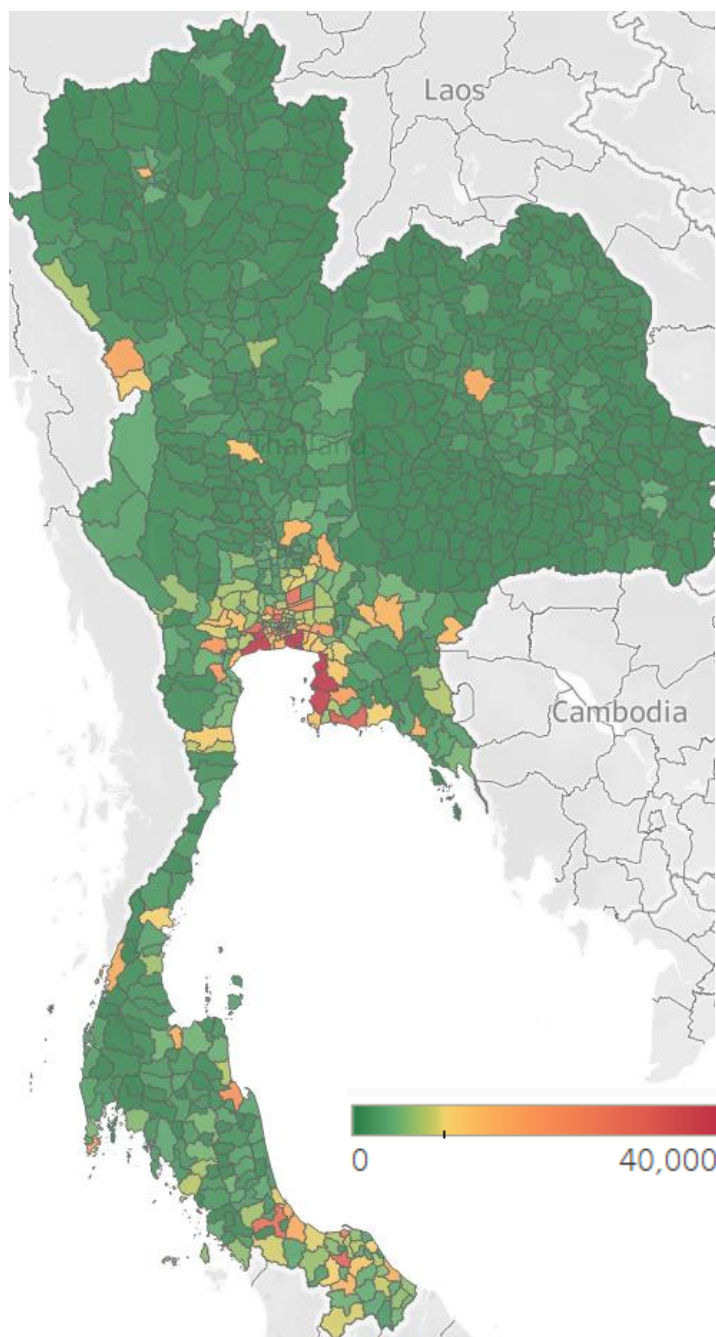
จากการทดสอบแสดงให้เห็นว่ากลุ่มข้อมูลที่ 2 มีความแปรปรวนประจำอำเภอสูงกว่ากลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 มีจำนวนผู้ติดเชื้อเฉลี่ยน้อยกว่ากลุ่มข้อมูลที่ 1 ทำให้สามารถสรุปได้ว่ากลุ่มข้อมูลที่ 2 หรืออำเภอที่มีค่า magnitude ใกล้เคียง -1 แสดงถึงอำเภอที่มีจำนวนผู้ติดเชื้อน้อยกว่าบริเวณรอบข้าง และยังแสดงให้เห็นว่ากลุ่มข้อมูล 3 มีความแปรปรวนประจำอำเภอสูงกว่ากลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 มีจำนวนผู้ติดเชื้อเฉลี่ยมากกว่ากลุ่มข้อมูลที่ 1 ทำให้สามารถสรุปได้ว่ากลุ่มข้อมูลที่ 3 หรืออำเภอที่มีค่า magnitude ใกล้เคียง 1 แสดงถึงอำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง

### 4.3 การสร้าง Heatmap ที่ได้จากค่า magnitude ประจำอำเภอ

เป็นขั้นตอนในการสร้างภาพ heatmap สำหรับนำไปวิเคราะห์เพื่อแปลผลนำไปต่อยอดในการควบคุมการระบาดต่อไป โดยจะสร้าง heatmap จากค่า magnitude ประจำอำเภอซึ่งมีค่าเพียงชุดเดียวที่ใช้อธิบายถึงรูปแบบการระบาด ได้ผลลัพธ์ดังภาพที่ 4.9 จากภาพจะเห็นได้ว่าสามารถแสดงข้อมูลเชิงลึกซึ่งอธิบายถึงการระบาดได้มากกว่า ภาพ heatmap ที่สร้างจากจำนวนผู้ติดเชื้อสะสมเพียงอย่างเดียวดังภาพที่ 4.10



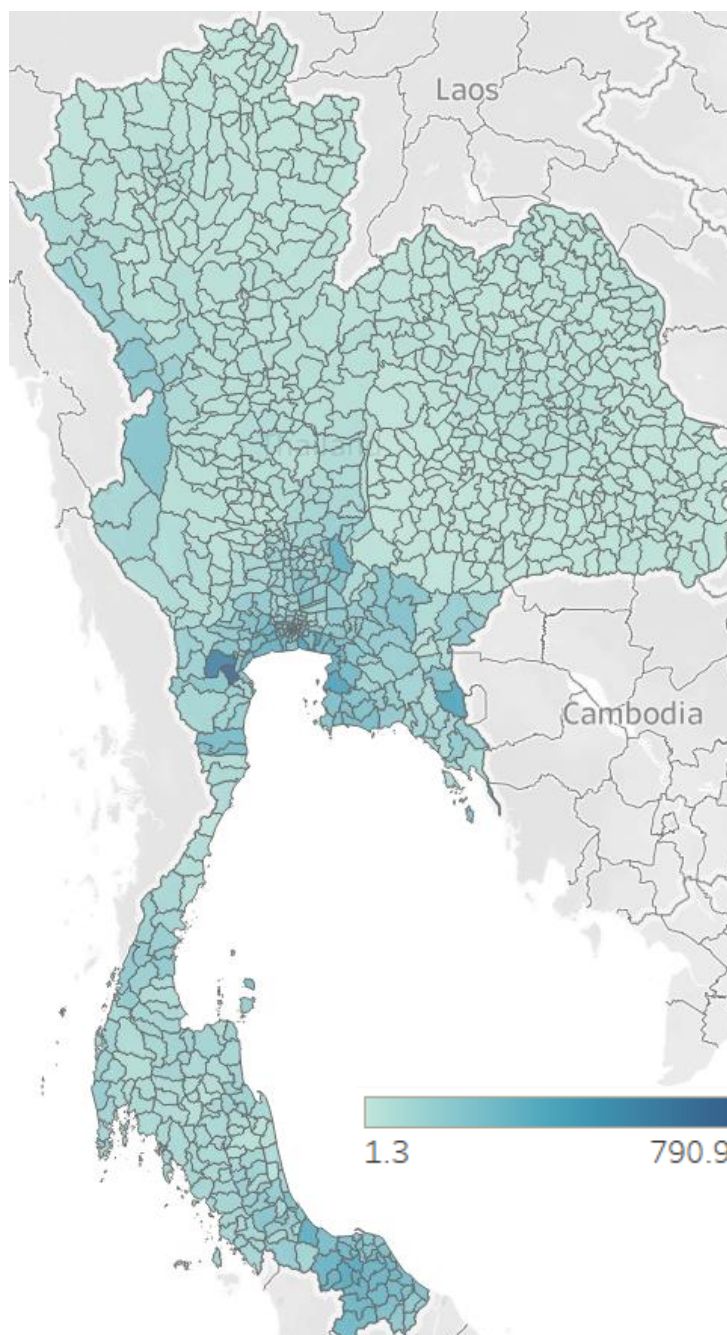
ภาพที่ 4.9: heatmap ที่สร้างจาก magnitude ประจำอำเภอ



ภาพที่ 4.10: heatmap ที่สร้างจากจำนวนผู้ติดเชื้อสะสม

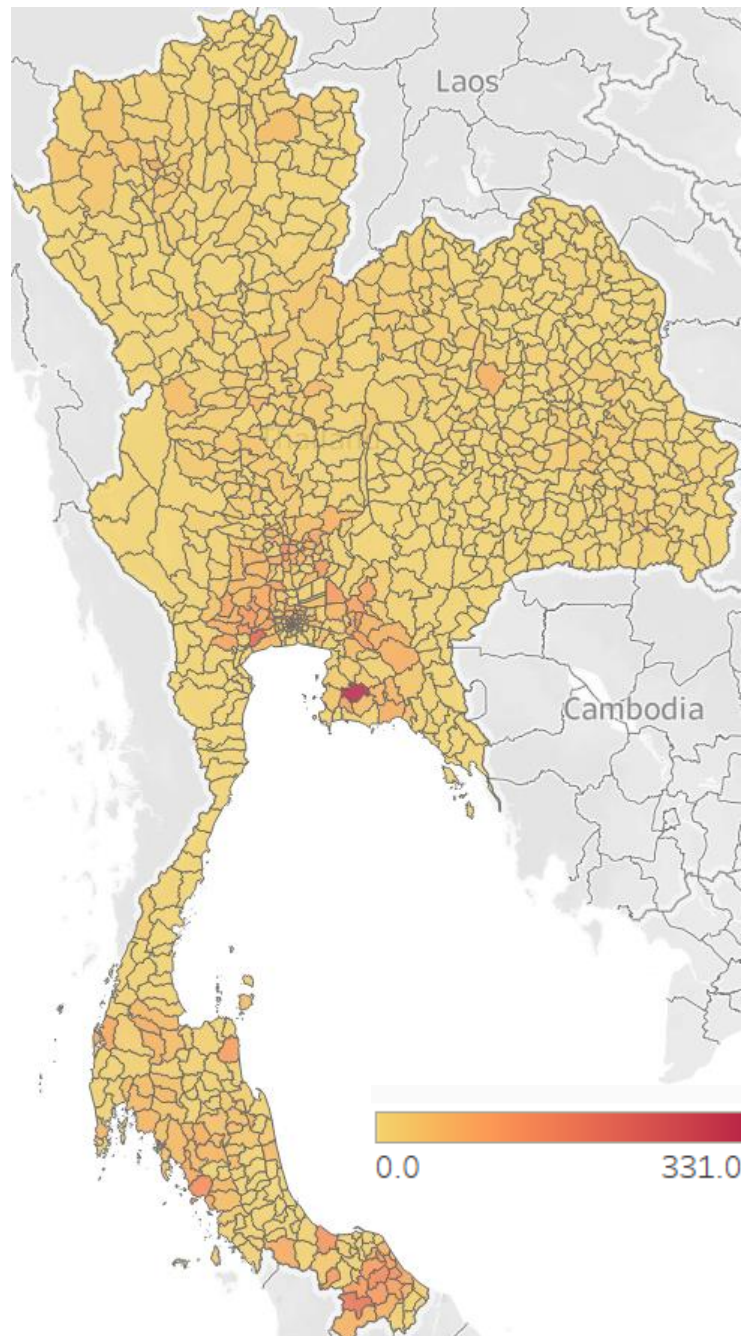
เมื่อนำไปเปรียบเทียบกับผลลัพธ์จากวิธีการของ Yang Li และ Gonzalo Mateos ซึ่งได้ผลลัพธ์ออกเป็นค่า magnitude 2 ชุดที่อธิบายถึงรูปแบบการระบาดดังภาพที่ 4.11 และ 4.12 โดยหากจะต้องการสรุปว่าอำเภอมีการระบาดในลักษณะใด จะต้องสรุปจากค่าทั้งสองค่า คือ magnitude ความถี่ต่ำ และ magnitude ความถี่สูง ซึ่งค่าทั้งสองค่านี้ตั้งอยู่บนช่วงค่าที่ต่างกันทำให้สรุปรูปแบบการระบาดในบางอำเภอไม่ชัดเจน ตัวอย่างเช่น อำเภอในบริเวณจังหวัดปัตตานีซึ่งทั้ง magnitude ความถี่ต่ำมีค่าสูง และ magnitude ความถี่สูงมีค่าสูง ทำให้สรุปได้ยากกว่าอำเภอในบริเวณดังกล่าวมี

จำนวนผู้ติดเชื้อใกล้เคียงใกล้เคียงรอบข้าง หรือแตกต่างจากรอบข้าง ซึ่งวิธีการที่นำเสนอสามารถแสดงรูปแบบการระบาดได้อย่างชัดเจน เนื่องจากค่า magnitude มีชุดเดียวทำให้การระบุรูปแบบการระบาดของแต่ละอำเภอได้อย่างชัดเจน ดังในภาพที่ 4.9 สามารถระบุรูปแบบการระบาดของอำเภอในบริเวณจังหวัดปัตตานีได้อย่างชัดเจน



ภาพที่ 4.11: heatmap ที่สร้างจาก magnitude ที่สร้างจากสัญญาณความถี่ต่ำประจำอำเภอ





ภาพที่ 4.12: heatmap ที่สร้างจาก magnitude ที่สร้างจากสัญญาณความถี่สูงประจำอำเภอ

## บทที่ 5

### ข้อสรุปและข้อเสนอแนะ

ในบทนี้จะกล่าวถึง สรุปผลวิจัยในการประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของโรคติดเชื้อโควิด-19 ในประเทศไทย และข้อเสนอแนะ

#### 5.1 ข้อสรุป

งานวิจัยนี้มีวัตถุประสงค์ในการระบุรูปแบบการระบาดของ COVID-19 ในประเทศไทยจากการประมวลผลสัญญาณกราฟโดยใช้ข้อมูลสัญญาณเป็นจำนวนผู้ติดเชื้อรายสัปดาห์ในแต่ละอำเภอ จำนวน 928 อำเภอ รวมทั้งสิ้น 39 สัปดาห์ และสร้างกราฟจากข้อมูลระยะห่างระหว่างอำเภอ โดยได้นำวิธีการของ Yang Li และ Gonzalo Mateos มาประยุกต์ใช้ โดยปัญหาที่พบจากวิธีการเดิมคือ ผลลัพธ์ที่ได้มีค่า 2 ชุดซึ่งแสดงถึงรูปแบบการระบาดทำให้การแปลผลเพื่อสรุปรูปแบบการระบาดในอำเภอทำได้ไม่ชัดเจน จึงได้เสนอวิธีการประมวลผลสัญญาณกราฟเพื่อระบุการระบาดโดยใช้สัญญาณในช่วงความถี่ต่ำเพียงอย่างเดียว โดยใช้หลักการที่ว่าผลลัพธ์ที่ได้เปรียบเสมือนการทำ localized averaging มาสร้างเป็น magnitude ค่าชุดเดียวที่แสดงถึงรูปแบบการระบาดที่แตกต่างกัน

จากผลการทดลองพบว่าวิธีการที่เสนอสามารถระบุรูปแบบการระบาดได้ โดยสามารถระบุรูปแบบการระบาดได้ 3 รูปแบบคือ อำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับบริเวณรอบข้าง, อำเภอที่มีจำนวนผู้ติดเชื้อน้อยกว่าบริเวณรอบข้าง และอำเภอที่มีจำนวนผู้ติดเชื้อมากกว่าบริเวณรอบข้าง ซึ่งสามารถระบุการระบาดได้มากกว่าวิธีการเดิมที่สามารถระบุการระบาดได้ 2 รูปแบบคือ อำเภอที่มีจำนวนผู้ติดเชื้อใกล้เคียงกับบริเวณรอบข้าง และอำเภอที่มีจำนวนผู้ติดเชื้อแตกต่างจากบริเวณรอบข้าง

จากผลการทดลองยังพบว่าวิธีการที่เสนอสามารถสรุปการระบาดในแต่ละอำเภอได้ชัดเจนกว่าวิธีการเดิม เนื่องจากค่าที่อธิบายถึงรูปแบบการระบาดหรือ magnitude เป็นค่าเพียงชุดเดียวซึ่งสร้างมาจากสัญญาณความถี่ต่ำ แตกต่างจากวิธีการเดิมซึ่งมีค่าอธิบายการระบาด 2 ชุด คือ magnitude ความถี่ต่ำ สร้างมาจากสัญญาณความถี่ต่ำ และ magnitude ความถี่สูง สร้างมาจากสัญญาณความถี่สูง

#### 5.2 ข้อเสนอแนะ

จากการทดลองประมวลผลสัญญาณกราฟเพื่อระบุรูปแบบการระบาดของ COVID-19 ในประเทศไทย ผู้วิจัยเห็นว่าควรมีการเพิ่มข้อเสนอดังต่อไปนี้

1. การกำหนดค่า magnitude สำหรับระบุการระบาดอาจสามารถเปลี่ยนการคำนวณเพื่อให้ได้ค่าที่ถูกต้องตามความหมายที่ต้องการมากขึ้น

2. การทดสอบยืนยันความถูกต้องในการระบุรูปแบบการระบาดสามารถใช้วิธีการทดสอบอื่นเพื่อเพิ่มความถูกต้องของผลลัพธ์มากขึ้น
3. การกำหนดค่า magnitude ประจำสัปดาห์สามารถคำนวณจากวิธีการอื่นนอกเหนือจากการใช้ความต่างระหว่างสัญญาณดั้งเดิมและสัญญาณที่ผ่านการกรองได้

## รายการอ้างอิง

- [1] Wijitbusaba Marome and Rajib Shaw. “COVID-19 Response in Thailand and Its Implications on Future Preparedness”, Int. J. Environ. Res. Public Health 2021, Vol. 18(3), Article 1089, 2021.
- [2] Kraichat Tantrakarnapaa and Bhophkrit Bhopdhornangkulb. “Challenging the Spread of COVID-19 in Thailand”, One Health and COVID-19, Vol. 11, Article 100173, 2020.
- [3] K. Elasmaoui and Y. Chawki. "Using X-ray Images and Deep Learning for Automated Detection of Coronavirus Disease", Journal of Biomolecular Structure and Dynamics, pp. 1-22, 2020.
- [4] Bowen Wang, Yanjing Sun, Trung Q. Duong, Long D. Nguyen and Lajos Hanzo. “Risk-Aware Identification of Highly Suspected COVID-19 Cases in Social IoT: A Joint Graph Theory and Reinforcement Learning Approach”, IEEE Access, Vol.8, pp. 115655–115661, 2020.
- [5] Yang Li and Gonzalo Mateos. “Graph Frequency Analysis of COVID-19 Incidence to Identify County-Level Contagion Patterns in the United States”, 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3230-3234, 2021.
- [6] Ljubiša Stanković, Miloš Daković, and Ervin Sejdić. “Vertex-Frequency Analysis of Graph Signals”, Signals and Communication Technology, Springer-Nature, 2019.
- [7] Isarapong Eksinchol. “Monitoring the COVID-19 Situation in Thailand”, 2020 1st International Conference on Big Data Analytics and Practices (IBDAP), pp 1-6, 2020.



- [8] Leah Goldsberry, Weiyu Huang, Nicholas F. Wymbs, Scott T. Grafton, Danielle S. Bassett and Alejandro Riberio. “Brain Signal Analytics from Graph Signal Processing Perspective”, 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 851 - 855, 2017.
- [9] Ljubiša Stanković, Danilo Mandić, Miloš Daković, Ilya Kisil, Ervin Sejdić and Anthony G. Constantinides. “Understanding the Basis of Graph Signal Processing via an Intuitive Example-Driven Approach”, IEEE Signal Processing Magazine, Vol.36, pp. 133 – 145, 2019.

ภาคผนวก

## ภาคผนวก ก

### แบบเสนอหัวข้อโครงการ รายวิชา 2301399 Project Proposal ปีการศึกษา 2564

ชื่อโครงการ (ภาษาไทย)	การสร้างกราฟเพื่อวิเคราะห์ความถี่กราฟสำหรับข้อมูลผู้ติดเชื้อ โควิด-19 รายวันในประเทศไทย
ชื่อโครงการ (ภาษาอังกฤษ)	Graph construction to analyse graph frequency for daily COVID-19 infected data in Thailand
อาจารย์ที่ปรึกษา	1. ผู้ช่วยศาสตราจารย์ ดร.อาทร เหลืองสดใส 2. อาจารย์ ดร.พรรพณ์ ปณิธานะรักษ์
ผู้ดำเนินการ	นาย ศรัณย์พัทธ ศุจินันท์ชยะกูร เลขประจำตัวนิสิต 6133758323 สาขาวิชา วิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์และวิทยาการ คอมพิวเตอร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

#### หลักการและเหตุผล

การระบาดของ COVID-19 เป็นปัญหาที่กระทบต่อประเทศไทยในทุกด้าน ซึ่งการควบคุมการระบาดได้รวดเร็วจะช่วยลดความเสียหายที่เกิดขึ้นได้อย่างมีประสิทธิภาพที่สุด จากการศึกษาของ Wijitbusaba Marome and Rajib Shaw [1] พบว่าประเทศไทยสามารถควบคุมการระบาดของ COVID-19 ได้อย่างมีประสิทธิภาพในการระบาดระลอกแรก และลดความเสียหายที่อาจเกิดขึ้นได้เป็นอย่างดี แต่ในการระบาดระลอกที่สามประเทศไทยไม่สามารถควบคุมได้เหมือนการระบาดระลอกแรก รวมทั้งการกลายพันธุ์ของเชื้อที่ทำให้การระบาดเป็นไปอย่างรวดเร็วมากขึ้น ส่งผลให้การควบคุมการระบาดเป็นไปได้ยากกว่าเดิม

เพื่อให้การควบคุมการระบาดมีประสิทธิภาพมากขึ้น เครื่องมือต่างๆ ที่ช่วยในการวิเคราะห์การระบาดหรือคาดการณ์การระบาดของ COVID-19 จึงถูกพัฒนาและนำมาใช้ โดยเครื่องมือที่มีการพัฒนาขึ้นในประเทศไทยส่วนมากจะเป็นการวิเคราะห์การระบาดจากข้อมูลดิบบางอย่างที่เก็บมา เช่น การนำตัวเลขผู้ติดเชื้อไปคาดการณ์จำนวนผู้ติดเชื้อรายใหม่โดยใช้ตัวแบบจำลอง SEIR [7] หรือการนำตัวแบบจำลอง SEIR ไปใช้ในการหาความสัมพันธ์การระบาดของ COVID-19 กับอุณหภูมิ [2] ซึ่งการ

วิเคราะห์ที่กล่าวมาไม่ได้มีการนำข้อมูลด้านอื่นมาศึกษาวิเคราะห์เพื่อหารูปแบบการระบาดร่วมด้วย เช่น ข้อมูลรับรู้เชิงพื้นที่ (spatial awareness) หรือ จำนวนประชากรในแต่ละพื้นที่

เครื่องมือที่ศึกษาและพัฒนาสำหรับช่วยในการควบคุมการระบาด COVID-19 มีด้วยกันหลายประเภท เช่น การใช้การเรียนรู้ของเครื่องในการระบุผู้ติดเชื้อจากภาพ x-ray ในปอด [3] การระบุความเสี่ยงที่จะติดเชื้อ COVID-19 จากการใช้ reinforcement learning ใน joint graph [4] ปัญหาที่เกิดขึ้นจากการนำเครื่องมือเหล่านี้มาใช้ในประเทศไทย คือ ข้อมูลที่มีอยู่ในประเทศไทยนำไปปรับใช้ได้ยาก เนื่องจากข้อมูลที่เกี่ยวข้องไม่ได้มีการเก็บข้อมูลไว้อย่างถูกต้องครบถ้วน หรือข้อมูลที่เกี่ยวข้องไม่มีการเผยแพร่ต่อสาธารณะหรือจำกัดการเข้าถึง ดังนั้นการสร้างเครื่องมือที่ซับซ้อนเพื่อช่วยในการควบคุมการระบาด COVID-19 ในประเทศไทยจึงเป็นไปได้ยาก

จากงานของ Yang Li and Gonzalo Mateos [5] ที่ได้มีการประยุกต์นำ Graph signal processing ไปศึกษา Graph frequencies เพื่อนำไปวิเคราะห์รูปแบบการระบาดของ COVID-19 ในประเทศสหรัฐอเมริกา เริ่มจากนำข้อมูลจำนวนผู้ติดเชื้อ COVID-19 รายวันในระดับ county มาสร้างเป็นสัญญาณ และสร้างกราฟจากข้อมูล commute flow จากนั้นทำการวิเคราะห์ความถี่กราฟ เริ่มจากแปลงกราฟ commute flow ให้เป็น Laplacian กราฟ ทำ graph Fourier transform ได้ผลลัพธ์คือ eigenvector และ eigenvalue ซึ่งแสดงถึงความถี่กราฟ จากนั้นทำการแบ่งช่วงความถี่ของ graph signal จากการทำ graph filtering เริ่มจากการสร้าง filter มีด้วยกัน 2 แบบคือ low-pass filter และ high-pass filter ซึ่งสร้างจากการเลือก eigenvector ที่ค่า eigenvalue ของมันอยู่ในช่วง 20% ที่น้อยที่สุดและช่วง 20% ที่มากที่สุดตามลำดับ หลังจากนั้นทำการกรองสัญญาณด้วย filter ที่สร้าง จะได้ผลลัพธ์เป็นสัญญาณที่ถูกแบ่งออกมาอยู่ในช่วงความถี่ที่กำหนด นั่นคือ low-pass หรือสัญญาณที่มีการเปลี่ยนแปลงน้อยเมื่อเทียบกับจุดในบริเวณรอบข้าง และ high-pass หรือสัญญาณที่มีการเปลี่ยนแปลงมากเมื่อเทียบกับจุดในบริเวณรอบข้าง โดยสามารถนำผลที่ได้จากการแบ่งช่วงความถี่ของ graph signal นี้ไปวิเคราะห์การระบาด COVID-19 ในระดับ county เพื่อพิจารณาว่าบริเวณใดมีการระบาดที่ต่อเนื่องกันเป็นรูปแบบเดียวกันเมื่อเทียบกับบริเวณรอบข้าง (low-pass regions) และบริเวณใดมีการระบาดในลักษณะที่แตกต่างกันเมื่อเทียบกับบริเวณรอบข้าง (high-pass regions) ซึ่งทางผู้วิจัยเห็นว่าผลที่ได้จากการวิเคราะห์ข้อมูลที่หลากหลาย แสดงให้เห็นถึงการระบาด COVID-19 ในเชิงลึกมากกว่าผลที่ได้จากการวิเคราะห์เพียงข้อมูลผู้ติดเชื้อรายวันอย่างเดียว ซึ่งผลลัพธ์เชิงลึกนี้สามารถนำไปประยุกต์เป็นข้อมูลเพิ่มเติมสำหรับการประกอบการตัดสินใจในการวางแผนควบคุมการระบาด ทั้งยังสามารถนำไปปรับพัฒนาให้ใช้ได้จริงในประเทศไทยอีกด้วย

จากการศึกษาในข้างต้น ผู้วิจัยมีความสนใจในการทำงานของ Yang Li and Gonzalo Mateos มาต่อยอดเพื่อพัฒนาเครื่องมือสำหรับวิเคราะห์การระบาด COVID-19 ในประเทศไทย

อย่างไรก็ตามงานของ Yang Li ได้มีการนำเอาข้อมูล commute flow มาสร้างเป็นกราฟสำหรับนำเอาไปวิเคราะห์ความถี่ด้วย ซึ่งข้อมูลในส่วนนี้ไม่ได้มีการจัดเก็บไว้หรือมีการเผยแพร่ให้เป็นสาธารณะในประเทศไทย ทำให้ผู้วิจัยต้องหาวิธีการสร้างกราฟแบบอื่นแทน จากการศึกษาใน [6] พบว่าการสร้างกราฟสำหรับนำไปวิเคราะห์ความถี่สามารถสร้างได้หลายแบบ ขึ้นอยู่กับการนิยามปัญหาให้สอดคล้องกับตัวข้อมูลและผลลัพธ์ที่สนใจ ทางผู้วิจัยจึงได้นำข้อมูลเชิงพื้นที่ได้แก่ ระยะห่างระหว่างอำเภอ และระยะเดินทางระหว่างอำเภอมาสร้างกราฟ และเปรียบเทียบผลลัพธ์ที่ได้เพื่อหากราฟที่เหมาะสมสำหรับการวิเคราะห์ความถี่เพื่อตรวจสอบรูปแบบการระบาด COVID-19 ในประเทศไทย

## วัตถุประสงค์

เพื่อพัฒนาเครื่องมือในการวิเคราะห์การระบาดเชิงพื้นที่ของ COVID-19 ในประเทศไทย โดยสร้างกราฟจากข้อมูลเชิงพื้นที่ วิเคราะห์ความถี่กราฟ และนำผลที่ได้มาจัดการกับข้อมูลผู้ติดเชื้อรายวันในแต่ละอำเภอ เพื่อหาว่าในบริเวณใกล้เคียงกัน บริเวณใดมีความถี่ผู้ติดเชื้อเป็นไปในรูปแบบเดียวกัน และบริเวณใดมีความถี่ผู้ติดเชื้อแตกต่างกัน ซึ่งผลลัพธ์ที่ได้ทั้งหมดจะถูกนำไปวิเคราะห์เพื่อตรวจสอบรูปแบบการระบาดในเชิงพื้นที่

## ขอบเขตของโครงการ

โครงการนี้ใช้ข้อมูลผู้ติดเชื้อรายวันในแต่ละอำเภอของประเทศไทย มีทั้งหมด 928 อำเภอ จาก 77 จังหวัด โดยนำข้อมูลมาจากเว็บไซต์ data.go.th ใช้ข้อมูลตั้งแต่ออนเริ่มการระบาดระลอกที่ 3 เป็นระยะเวลา 1 เดือน หรือข้อมูลตั้งแต่วันที่ 1 มีนาคม 2564 จนถึงวันที่ 1 ธันวาคม 2564 ผลลัพธ์ที่ได้จะเป็นข้อมูลสองชุดคือ ค่าความเป็น low-pass หรือค่าความเป็นต่อเนื่องกัน เมื่อเทียบกับบริเวณรอบข้าง และค่าความเป็น high-pass หรือค่าความไม่ต่อเนื่องกันเมื่อเทียบกับบริเวณรอบข้าง

## วิธีการดำเนินงาน

### แผนการดำเนินงาน

1. ศึกษาค้นคว้าทฤษฎีที่เกี่ยวข้อง
2. เตรียมข้อมูลผู้ติดเชื้อรายวันในแต่ละอำเภอ และทำความสะอาดข้อมูล
3. สร้างกราฟรูปแบบต่างๆ จากข้อมูล ระยะห่างระหว่างอำเภอ และระยะเดินทางระหว่างอำเภอ

3.1 สร้างกราฟโดยใช้ gaussian kernel โดยมี threshold เป็นค่าเฉลี่ยอำเภอที่อยู่ใกล้กัน



6. ประเมินผลลัพธ์ที่ได้จากการวิเคราะห์ความถี่กราฟ									
7. วิเคราะห์และอภิปรายผล									
8. จัดทำเอกสาร									

## ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ต่อผู้พัฒนา

1. ฝึกฝนและพัฒนาทักษะในการวางแผนและการทำงานเป็นขั้นตอน
2. ฝึกฝนและพัฒนาทักษะในการพัฒนาเครื่องมือสำหรับการวิเคราะห์ข้อมูลด้วยกราฟ
3. พัฒนาศักยภาพในการเรียนรู้ด้วยตนเอง

ประโยชน์ที่ได้จากโครงการ

1. ผู้ใช้ได้ข้อมูลเชิงลึกเกี่ยวกับการระบาดของ COVID-19 นอกเหนือจากแผนภาพจำนวนผู้ติดเชื้อรายวัน
2. ผู้ใช้สามารถนำข้อมูลรูปแบบการระบาดเชิงพื้นที่ไปประยุกต์ใช้ในการวางแผนควบคุมการระบาดได้ดียิ่งขึ้น เช่น การใช้เป็นเกณฑ์ประกอบในการแบ่งพื้นที่ความรุนแรงของการระบาด
3. สามารถนำผลการสร้างกราฟไปประยุกต์ต่อยอดในการพัฒนาเครื่องมืออื่นที่มีความใกล้เคียงกันได้

## อุปกรณ์และเครื่องมือที่ใช้

1. ฮาร์ดแวร์

คอมพิวเตอร์ที่มีรุ่นของหน่วยประมวลผลกลางไม่ต่ำกว่า Intel Core i3-9100F @3.60 GHz และความจุของหน่วยความจำไม่ต่ำกว่า 8 GB

2. ซอฟต์แวร์

2.1 Visual Studio Code เวอร์ชัน 1.61

2.2 wolfram mathematica เวอร์ชัน 12

2.3 ชุดคำสั่งในภาษา Python สำหรับการทำงานและประมวลผลข้อมูล เช่น numpy, pandas, scipy, matplotlib, wikipedia, pygsp

2.4 Tableau เวอร์ชัน 2020.3

## งบประมาณ

1. หูฟังไร้สาย	1 ชิ้น	3,730 บาท
2. เมาส์ไร้สาย	1 ชิ้น	1,520 บาท
		<b>รวม 5,250 บาท</b>

## เอกสารอ้างอิง

[1] Wijitbusaba Marome and Rajib Shaw. “COVID-19 Response in Thailand and Its Implications on Future Preparedness”, Int. J. Environ. Res. Public Health 2021, Vol. 18(3), pp. 1089, 2021.

[2] Kraichat Tantrakarnapaa and Bhophkrit Bhophdhornangkulb. “Challenging the spread of COVID-19 in Thailand”, Vol. 11, pp. 100173, 2020.

[3] K. Elasnoui and Y. Chawki. "Using X-ray Images and Deep Learning for Automated Detection of Coronavirus Disease", Journal of Biomolecular Structure and Dynamics, pp. 1-22, 2020.

[4] Bowen Wang, Yanjing Sun, Trung Q. Duong, Long D. Nguyen and Lajos Hanzo. “Risk-Aware Identification of Highly Suspected COVID-19 Cases in Social IoT: A Joint Graph Theory and Reinforcement Learning Approach”, IEEE Access, Vol.8, pp. 115655–115661, 2020.

[5] Yang Li and Gonzalo Mateos. “Graph Frequency Analysis of COVID-19 Incidence to Identify County-Level Contagion Patterns in the United States”, ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3230-3234, 2021

[6] Ljubiša Stanković, Miloš Daković, and Ervin Sejdić. “Vertex-Frequency Analysis of Graph Signals”, Signals and Communication Technology, Springer-Nature, 2019

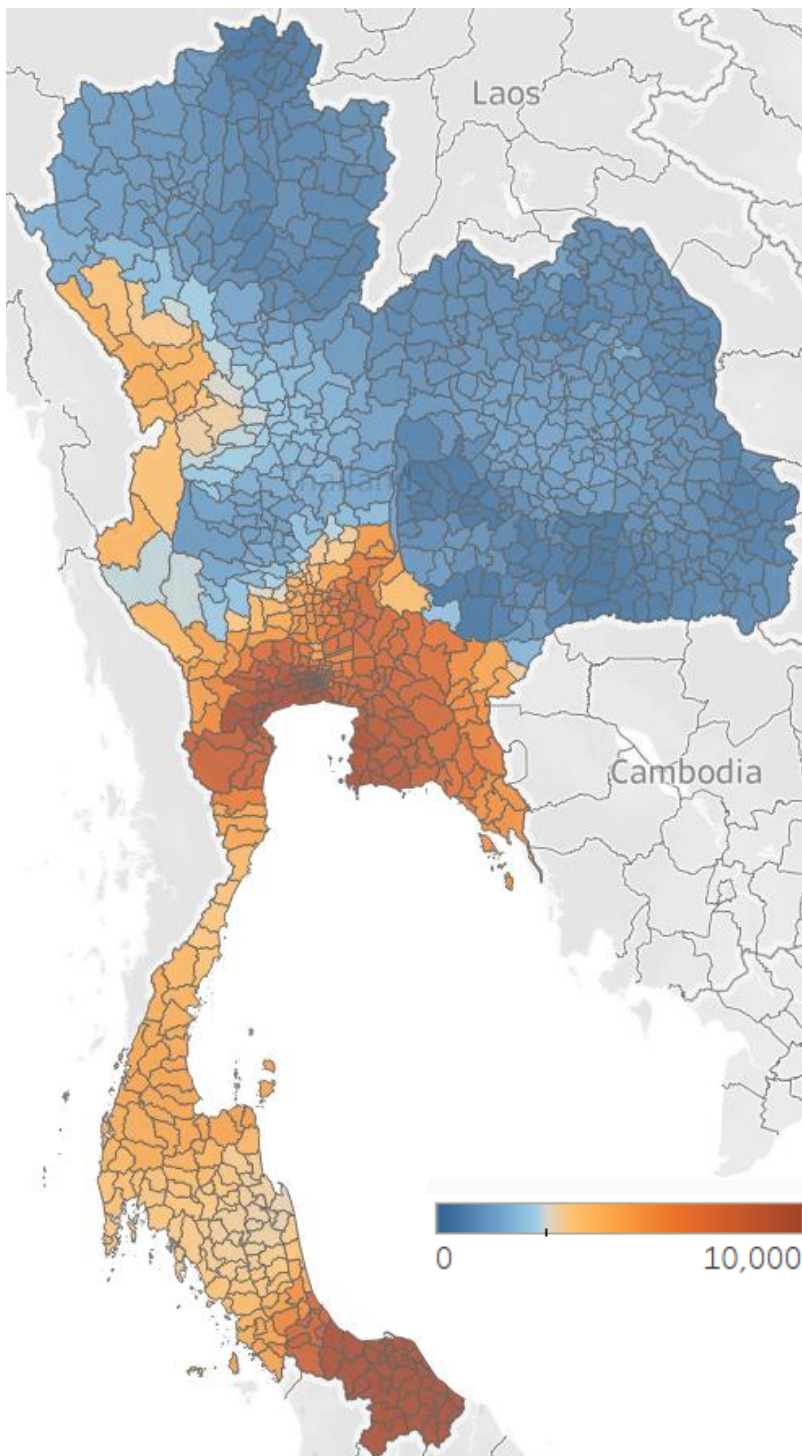
[7] Isarapong Eksinchol. “Monitoring the COVID-19 Situation in Thailand”, 2020 1st International Conference on Big Data Analytics and Practices (IBDAP), pp 1-6, 2020



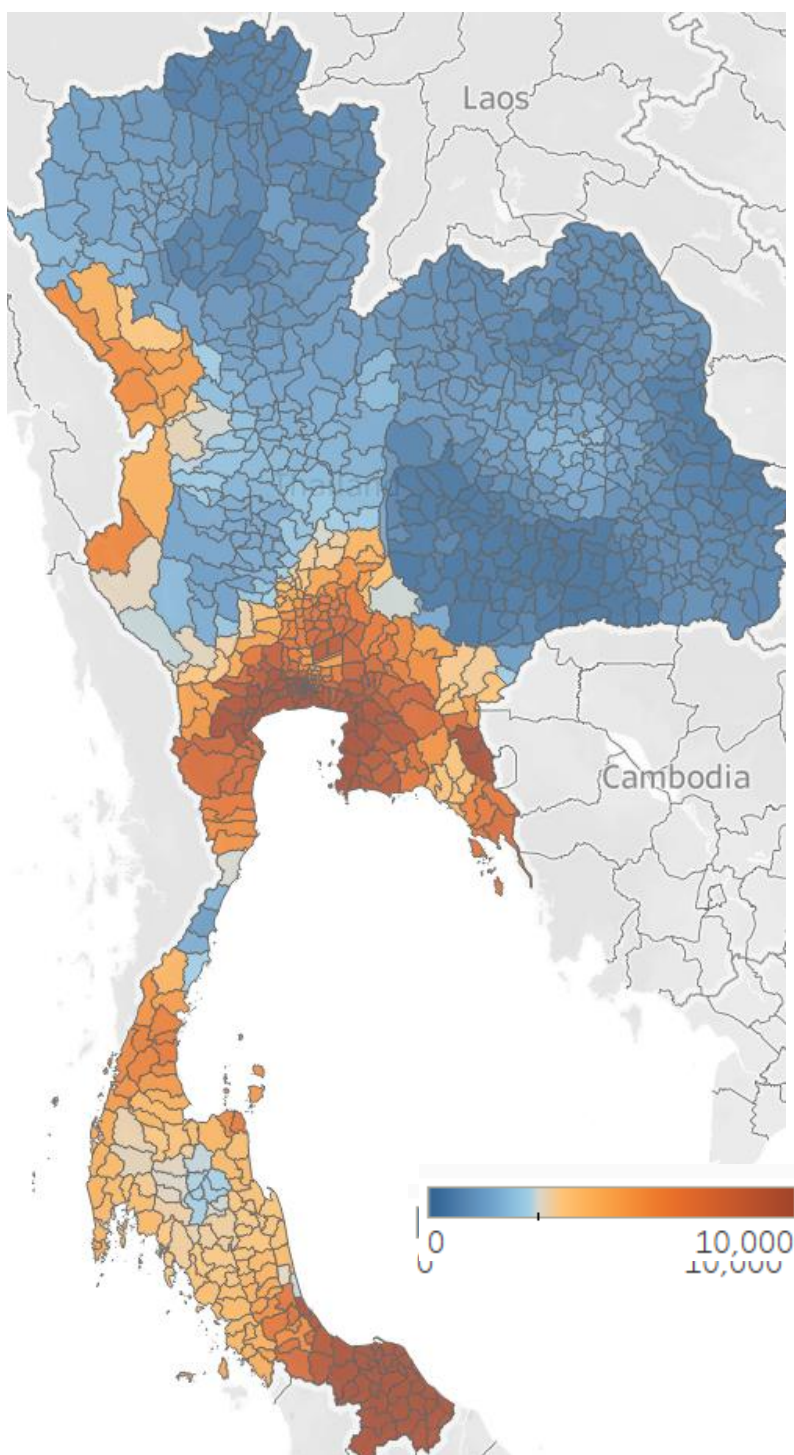
## ภาคผนวก ข

## ภาพผลลัพธ์สัญญาณที่ถูกปรับให้เรียบที่ได้ในแต่ละ filter

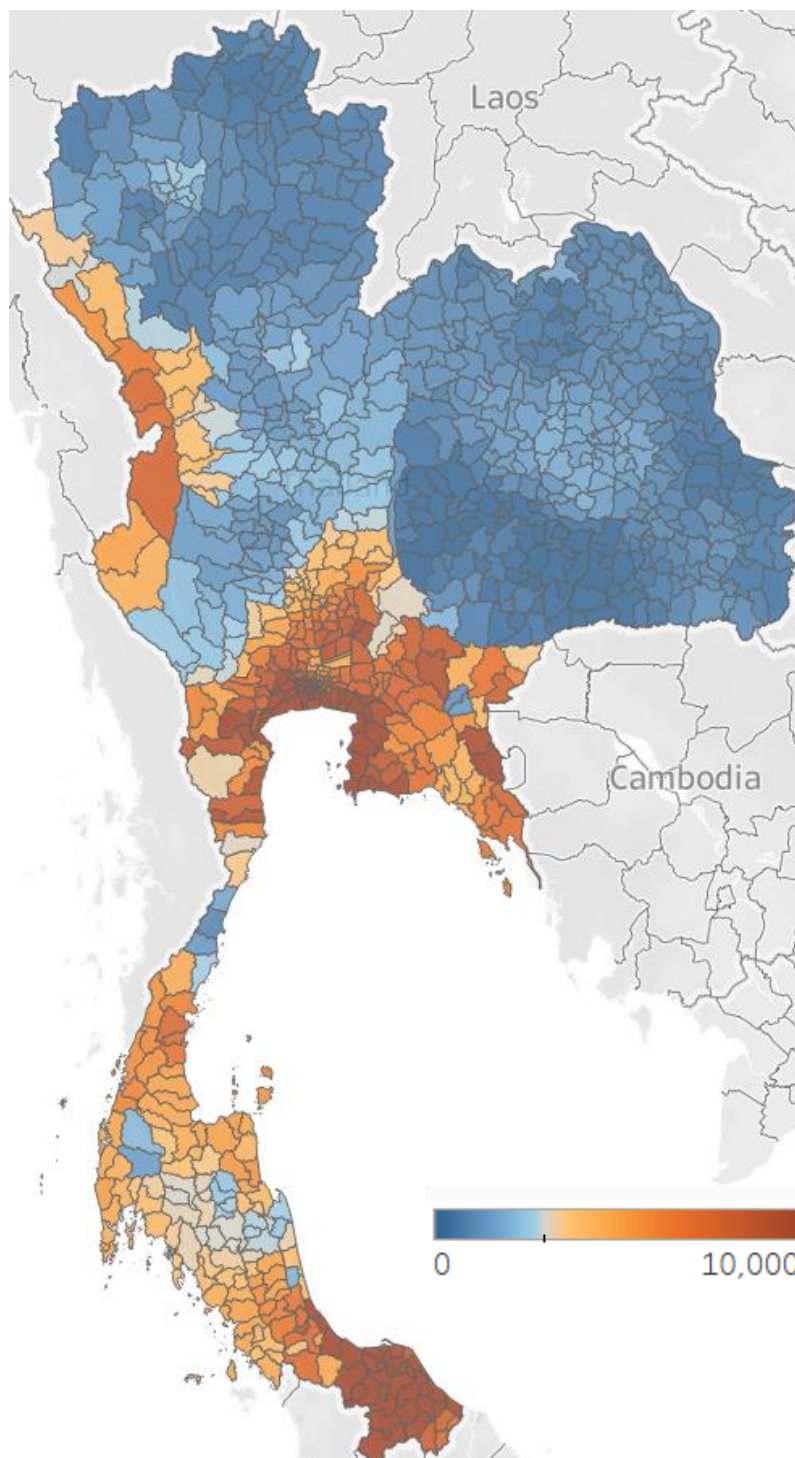
1. ภาพสัญญาณที่ถูกปรับให้เรียบด้วย filter ที่สร้างจากการเลือก eigenvectors มาด้วยปริมาณ 5% ของปริมาณ eigenvectors ทั้งหมด



2. ภาพสัญญาณที่ถูกปรับให้เรียงด้วย filter ที่สร้างจากการเลือก eigenvectors มาด้วย ปริมาณ 10% ของปริมาณ eigenvectors ทั้งหมด

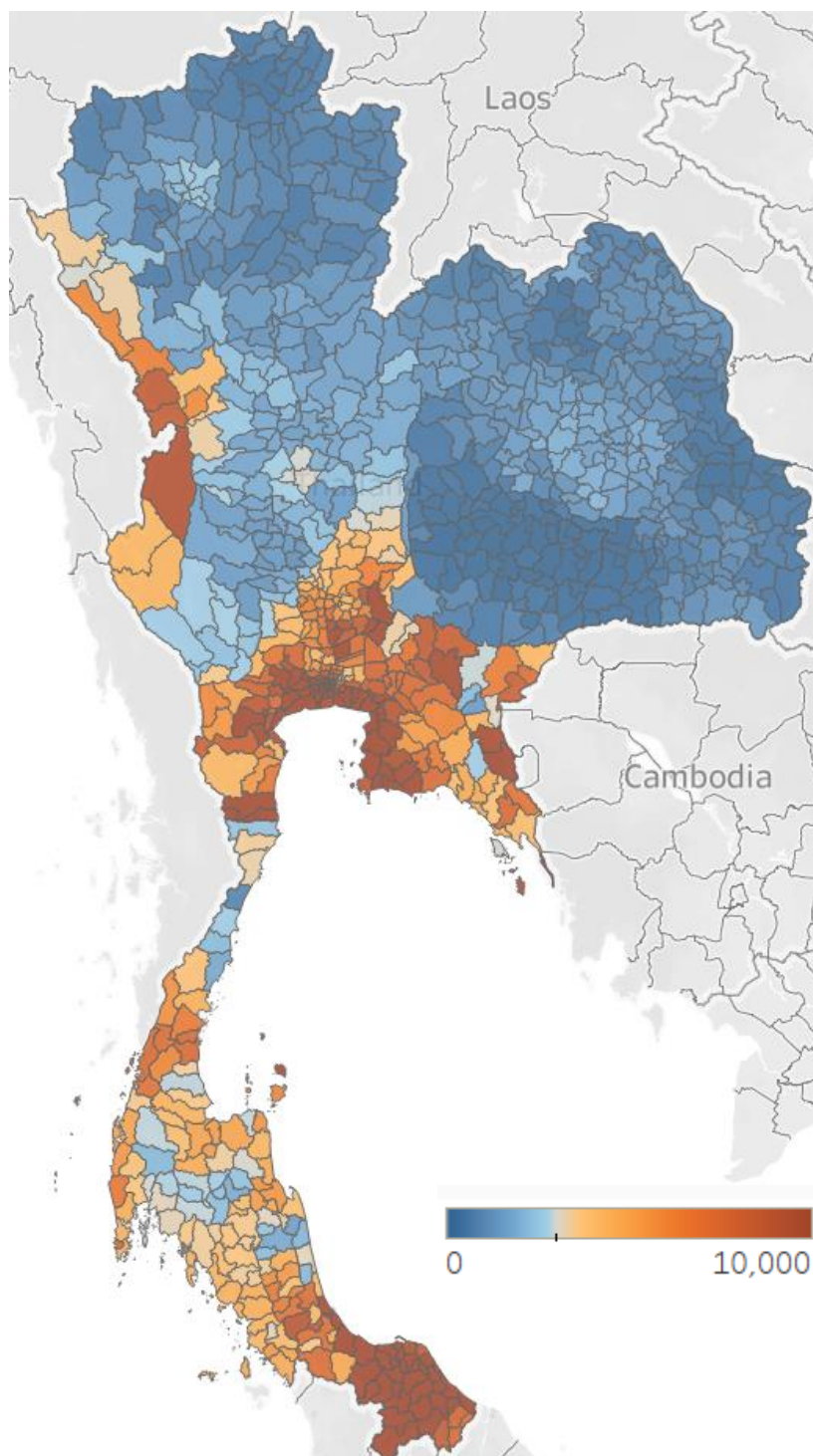


3. ภาพสัญญาณที่ถูกปรับให้เรียบด้วย filter ที่สร้างจากการเลือก eigenvectors มาด้วย ปริมาณ 20% ของปริมาณ eigenvectors ทั้งหมด

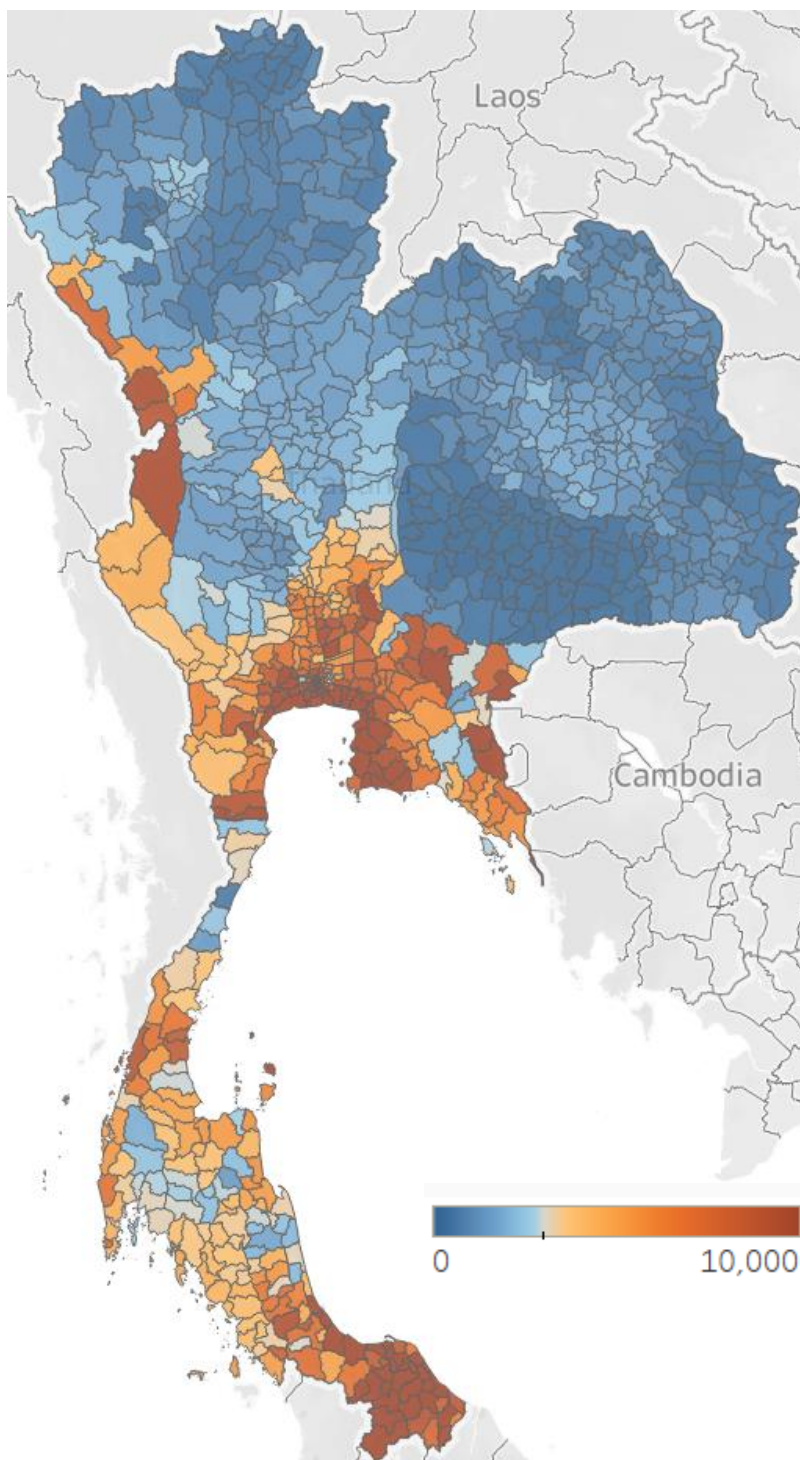




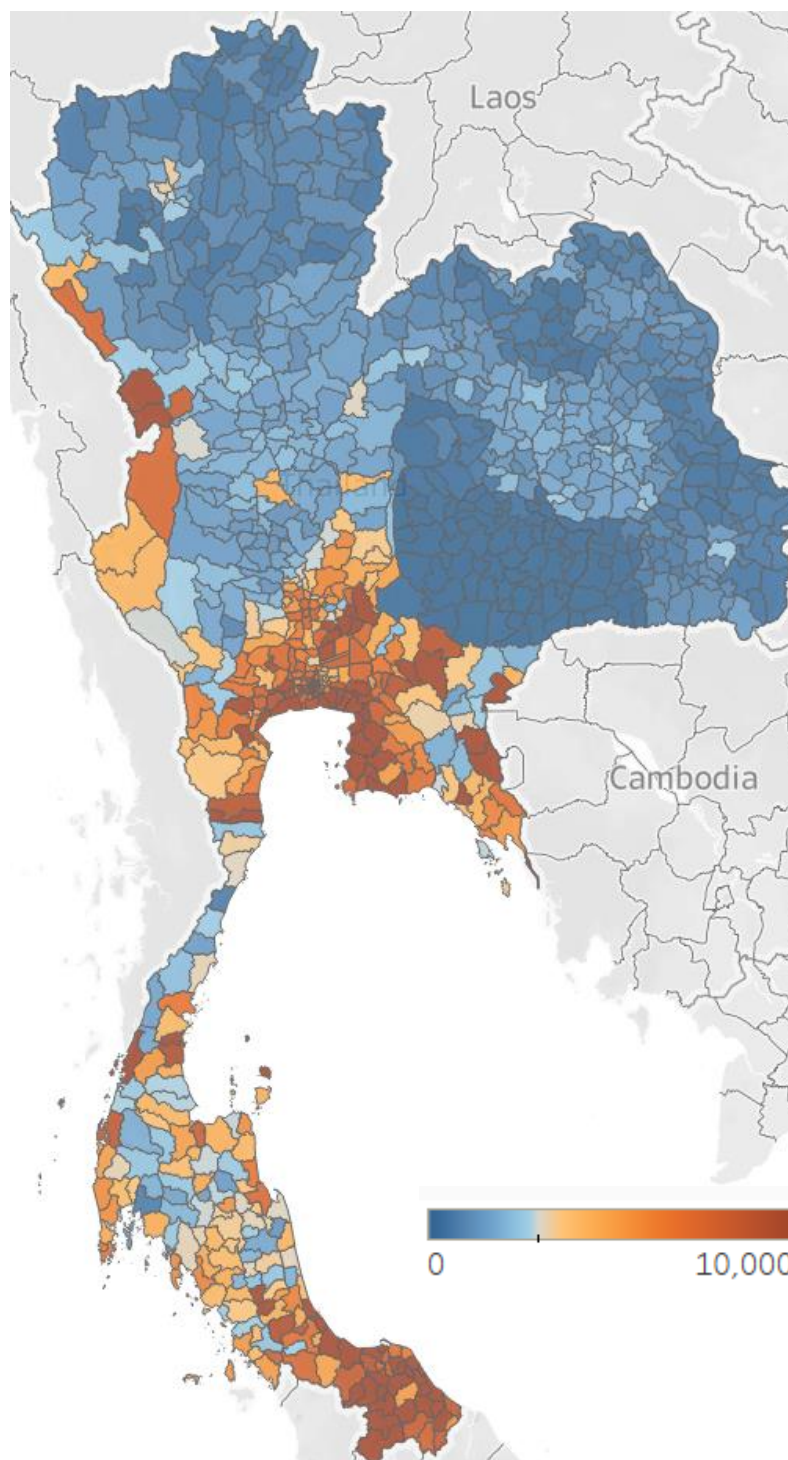
4. ภาพสัญญาณที่ถูกปรับให้เรียบด้วย filter ที่สร้างจากการเลือก eigenvectors มาด้วย ปริมาณ 30% ของปริมาณ eigenvectors ทั้งหมด



5. ภาพสัญญาณที่ถูกปรับให้เรียบด้วย filter ที่สร้างจากการเลือก eigenvectors มาด้วย ปริมาณ 40% ของปริมาณ eigenvectors ทั้งหมด

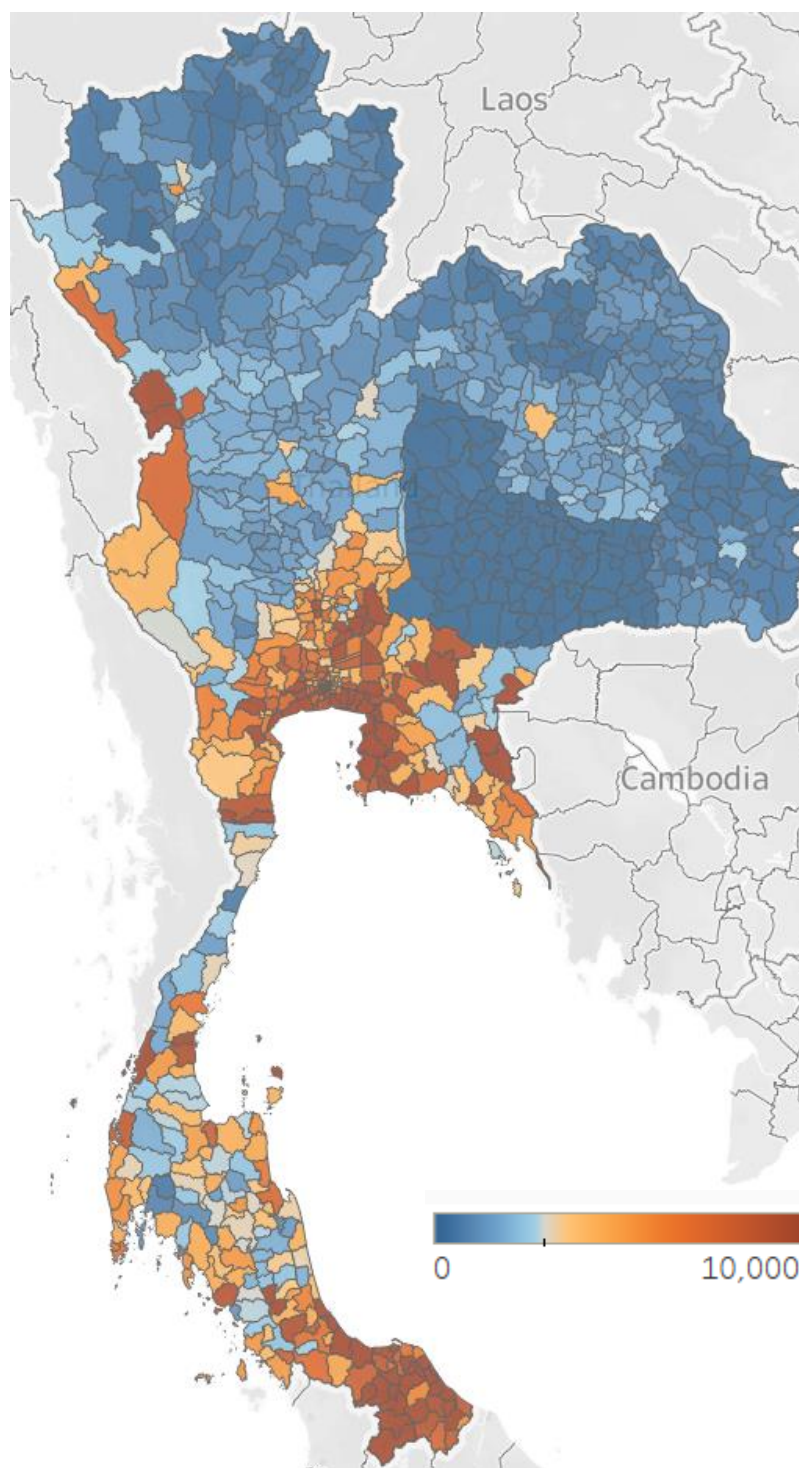


6. ภาพสัญญาณที่ถูกปรับให้เรียบด้วย filter ที่สร้างจากการเลือก eigenvectors มาด้วย ปริมาณ 80% ของปริมาณ eigenvectors ทั้งหมด





## 7. ภาพสัญญาณตั้งต้น



## ภาคผนวก ค

### ตัวอย่างส่วนของโปรแกรมที่ใช้ในการประมวลผลสัญญาณกราฟ

#### 1. ส่วนของการสร้างเมทริกซ์สัญญาณ

โดยทำการอ่านข้อมูลจำนวนผู้ติดเชื้อรายวันในแต่ละอำเภอที่ทำความสะอาดแล้ว สร้างเป็นสัญญาณรายวัน จากนั้นแปลงเป็นสัญญาณรายสัปดาห์ และ normalize เป็นจำนวนผู้ติดเชื้อต่อประชากร 100,000 คน

```
import pandas as pd

case = pd.read_excel(r'District_daily_clean.xlsx')
pop = pd.read_excel(r'District_pop.xlsx')

signal_daily = [[0 for day in range(273)] for district in range(928)]
N = case.shape[0]  #นับจำนวน case ทั้งหมดที่เกิดขึ้น
for i in range (N):
    for j in range (928):
        if pop.at[j,'District'] == case.at[i,'district_of_onset'] :
            if pop.at[j,'Province']== case.at[i,'province_of_onset']:
                posDis = j
                break
    signal_daily[posDis][case.at[i,'announce_date']-44287] += 1

signal_weekly = [[0 for week in range(39)] for district in range(928)]
for i in range(928):
    for j in range(39):
        signal_weekly[i][j] = round(sum(signal_daily[i][j*7:j*7+7]))

for i in range(928):
    for j in range(39):
        signal_weekly[i][j] = signal_weekly[i][j]*100000/pop.at[i,'Pop']
```



## 2. ส่วนของการสร้างกราฟ

เริ่มจากการอ่านข้อมูลระยะห่างระหว่างอำเภอซึ่งเป็นเมทริกซ์  $928 \times 928$  ถัดไปหาระยะห่างที่สั้นที่สุดของอำเภอที่กำลังพิจารณา จากนั้นกำหนดน้ำหนักของเส้นเชื่อมของอำเภอที่กำลังพิจารณาตามสมการ (5)

```
import pandas as pd

distance = pd.read_excel('distance.xlsx')
distance_arr = distance.to_numpy()

Graph = [[0 for column in range(928)] for row in range(928)]

for i in range(928):
    row = np.array(distance_arr[i])
    min_r = min(row[row != 0]) #หาระยะห่างที่สั้นที่สุดของอำเภอ i
    for j in range(928):
        if distance_arr[i][j] == 0 :
            continue
        else :
            if distance_arr[i][j] >= 4*min_r : Graph[i][j] = 0
            else :
                Graph[i][j] = 1-math.log(distance_arr[i][j]/min_r,4)
```

## 3. ส่วนของการปรับสัญญาณให้เรียบแบบในอุดมคติ และการกำหนดค่า magnitude

เริ่มจากการอ่านข้อมูลสัญญาณ และกราฟ จากนั้นแยกส่วนประกอบ eigenvector ของกราฟ สร้าง filter ตามสมการ (4) จากนั้นนำ filter ที่สร้างไปคูณกับสัญญาณได้สัญญาณที่ผ่านการกรอง สุดท้ายคำนวณ magnitude ประจำอำเภอตามสมการ (8)

```
import numpy as np
from pygsp import graphs
import pandas as pd
import math
```

```

W_dataframe = pd.read_excel('Graph.xlsx')
x_dataframe = pd.read_excel(r'signal_normalize.xlsx')
x_raw_dataframe = pd.read_excel(r'signal.xlsx')

x = x_dataframe.to_numpy()
x_raw = x_raw_dataframe.to_numpy()
W = W_dataframe.to_numpy()

G = graphs.Graph(W)
G.compute_laplacian()
G.compute_fourier_basis()

v = G.u
vt = np.transpose(v)

filter_30_hat = np.zeros((928,928))
sm = 0.3
for i in range(928):
    if i<=928*sm : filter_30_hat[i][i]=1
    else : filter_30_hat[i][i] = 0

filter_30 = np.matmul(np.matmul(v,filter_30_hat),vt)
xl = np.matmul(filter_30,x)

magnitude = [0 for district in range(928)]
for i in range(928):
    magnitude[i] = math.log((sum(x[i])+500)/(sum(abs(xl[i]))+500),2)
    if magnitude[i]>1 : magnitude[i] = 1
    if magnitude[i]<-1 : magnitude[i] = -1

```

#### 4. ส่วนของการสร้างกลุ่มข้อมูลสำหรับการทดสอบ

สร้างกลุ่มข้อมูลที่ 1 จากการเลือกอำเภอที่มีค่า magnitude ใกล้เคียง 0 ที่สุดทั้งจำนวนบวก และจำนวนลบ หรือก็คืออำเภอที่มีค่า magnitude กำลัง 2 มีค่าน้อยที่สุดมา 100 อำเภอ สร้างกลุ่มข้อมูลที่ 2 สร้างจากการเลือกอำเภอที่มีค่า magnitude น้อยที่สุดมา 100 อำเภอ และสร้างกลุ่มข้อมูลที่ 3 จากการเลือกอำเภอที่มีค่า magnitude สูงที่สุดมา 100 อำเภอ ผลลัพธ์ที่ได้จะเป็น index ของอำเภอในแต่ละกลุ่มข้อมูล

```
magnitude_square = [ magnitude[i]**2 for i in range (928)]
```

```
group_1 = np.argpartition(magnitude_square, 100)[:100]
```

```
group_2 = np.argpartition(magnitude, 100)[:100]
```

```
group_3 = np.argpartition(magnitude, -100)[-100:]
```

#### 5. ส่วนของการทดสอบ t-test ของค่าทดสอบความแปรปรวนประจำอำเภอ

เริ่มจากคำนวณความแปรปรวนประจำอำเภอของแต่ละอำเภอในแต่ละกลุ่มข้อมูลตามสมการ (10) ด้วยข้อมูลจำนวนผู้ติดเชื้อที่แท้จริง จากนั้นทำการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 และทำการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 ด้วย  $p\_value = 0.05$

```
from scipy.stats import t
```

```
def TV_r(i) :
```

```
    TV_r = 0
```

```
    for k in range(39):
```

```
        for j in range(928):
```

```
            if i != j and W[i][j]!=0:
```

```
                TV_r += W[i][j]*abs(math.log((x_raw[i][k]+1)/(x_raw[j][k]+1),2))
```

```
    return TV_r
```

```
var_district_G1 = []
```

```
var_district_G2 = []
```

```
var_district_G3 = []
```

```

for i in range(100):
    var_district_G1.append(TV_r[group_1[i]])
    var_district_G2.append(TV_r[group_2[i]])
    var_district_G3.append(TV_r[group_3[i]])

mean_G1 = np.mean(var_district_G1)
sd_G1 = np.std(var_district_G1)
mean_G2 = np.mean(var_district_G2)
sd_G2 = np.std(var_district_G2)
mean_G3 = np.mean(var_district_G3)
sd_G3 = np.std(var_district_G3)

t_value = (mean_G1-mean_G2)/(np.sqrt(sd_G1**2/100+sd_G2**2/100))
t_test = t.cdf(t_value,99)*2
result_var_1v2 = t_test<0.05

t_value = (mean_G1-mean_G3)/(np.sqrt(sd_G1**2/100+sd_G3**2/100))
t_test = t.cdf(t_value,99)*2
result_var_1v3 = t_test<0.05

```

#### 6. ส่วนของการทดสอบ t-test ของค่าทดสอบจำนวนผู้ติดเชื้อเฉลี่ยประจำอำเภอ

เริ่มจากคำนวณจำนวนผู้ติดเชื้อเฉลี่ยของแต่ละอำเภอในแต่ละกลุ่มข้อมูล โดยใช้จำนวนผู้ติดเชื้อที่แท้จริง จากนั้นทำการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 2 และทำการทดสอบระหว่างกลุ่มข้อมูลที่ 1 และกลุ่มข้อมูลที่ 3 ด้วย  $p\_value = 0.05$

```
avg_G1 = []
avg_G2 = []
avg_G3 = []

for i in range(100):
    avg_G1.append(np.average(x_raw[group_1[i]]))
    avg_G2.append(np.average(x_raw[group_2[i]]))
    avg_G3.append(np.average(x_raw[group_3[i]]))

mean_G1 = np.mean(avg_G1)
sd_G1 = np.std(avg_G1)
mean_G2 = np.mean(avg_G2)
sd_G2 = np.std(avg_G2)
mean_G3 = np.mean(avg_G3)
sd_G3 = np.std(avg_G3)

t_value = (mean_G1-mean_G2)/(np.sqrt(sd_G1**2/100+sd_G2**2/100))
t_test = t.cdf(t_value,99)*2
result_avg_1v2 = t_test<0.05

t_value = (mean_G1-mean_G3)/(np.sqrt(sd_G1**2/100+sd_G3**2/100))
t_test = t.cdf(t_value,99)*2
result_avg_1v3 = t_test<0.05
```

## 7. ส่วนของการสร้าง heat map

สร้างโดย tableau public นำข้อมูลรูปร่างในระดับอำเภอ หรือ adm2 จากเว็บไซต์ <https://data.humdata.org/dataset/cod-ab-tha> มาเชื่อมกับข้อมูล magnitude ประจำอำเภอ โดยเลือกเชื่อมด้วย attribute รหัสของอำเภอ

The screenshot shows the Tableau Public interface with the following components:

- Connections:** Lists two data sources: 'tha\_admbnda\_adm2\_rtsd\_20220121' (Spatial file) and 'mag\_sm\_0.3' (Microsoft Excel).
- Sheets:** Shows 'Sheet1' and a 'New Union' button.
- Data Source:** Displays the connection to 'tha\_admbnda\_adm2\_rtsd\_20220121.shp+ (Multiple...)'. A diagram shows this source connected to 'Sheet1'.
- Field List:** Shows the relationship between 'tha\_admbnda\_adm...' and 'Sheet1' with the operator '='. Fields include 'Adm2 Pcode' and 'Adm2 Pcode1'.
- Data Preview:** A table showing the first 13 rows of data. The columns are '#', 'Mag', and 'Adm2 Pcode1'.

#	Mag	Adm2 Pcode1
Sheet1	0.03618	TH1001
Sheet1	-0.41774	TH1002
Sheet1	0.01864	TH1003
Sheet1	-0.19083	TH1004
Sheet1	-0.76348	TH1005
Sheet1	0.30327	TH1006
Sheet1	0.90146	TH1007
Sheet1	0.60231	TH1008
Sheet1	-0.52719	TH1009
Sheet1	0.49848	TH1010
Sheet1	0.10663	TH1011
Sheet1	0.22425	TH1012
Sheet1	0.42810	TH1013

## ประวัติผู้เขียน



นาย ศรัณย์พัทธ์ ศุจินันท์ชยะกูร

รหัสนิสิต 6133758323

วันเดือนปีเกิด 2 สิงหาคม 2543

ภูมิลำเนา จังหวัดราชบุรี

กำลังศึกษาในสาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย