

## สารบัญ

	หน้า
บทที่ 1 บทนำ .....	3
1.1 หลักการและเหตุผล .....	3
1.2 วัตถุประสงค์ของโครงการ .....	3
1.3 ขอบเขตของโครงการ .....	3
1.4 ขั้นตอนการดำเนินงาน .....	5
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	5
1.6 อุปกรณ์และเครื่องมือที่ใช้ในการดำเนินงาน .....	6
1.7 แผนการดำเนินงาน .....	6
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	8
2.1 ทฤษฎีที่เกี่ยวข้อง .....	8
2.1.1 คาร์บอนไดออกไซด์ .....	8
2.1.2 Google Colab .....	8
2.1.3 การเตรียมข้อมูลสำหรับการจำแนก .....	8
2.1.4 การทำเหมืองข้อมูล (data mining) .....	8
2.1.5 Time Series .....	8
2.1.6 วิธีการคัดเลือกตัวแปรอิสระแบบการกำจัดแบบถอยหลัง (Backward eliminate) .....	8
2.1.7 อัลกอริทึม Decision tree(ต้นไม้ตัดสินใจ) .....	8
2.1.8 อัลกอริทึม Random Forest .....	8
2.1.9 อัลกอริทึม Linear Regression .....	8
2.1.10 Support Vector Machine (SVM) .....	8
2.1.11 การวัดประสิทธิภาพ .....	8
2.1.12 การแสดงข้อมูลด้วยภาพ .....	8
2.2 งานวิจัยที่เกี่ยวข้อง .....	8

บทที่ 3 วิธีดำเนินโครงการ.....	8
3.1 แผนการดำเนินงาน.....	8
เอกสารอ้างอิง .....	19
ภาพประกอบที่ 1 แผนผังขั้นตอนการดำเนินงาน.....	5
ภาพประกอบที่ 2 Google Colaboratory .....	8
ภาพประกอบที่ 3 ต้นไม้ตัดสินใจในการป้องกันภาวะหัวใจวาย .....	8
ภาพประกอบที่ 4 การแตกกิ่งของต้นไม้ .....	8
ภาพประกอบที่ 5 หลักการทำ Random Forest.....	8
ภาพประกอบที่ 6 Positive Linear Relationship.....	8
ภาพประกอบที่ 7 Negative Linear Relationship .....	8
ภาพประกอบที่ 8 No Apparent Linear Relationship.....	8
ภาพประกอบที่ 9 ข้อมูลขนาด 2 มิติซึ่งจุดจำแนกออกเป็น 2 Class .....	8
ภาพประกอบที่ 10 ตัวอย่างการปรับค่า Parameter C ที่มีผลต่อขนาดของเส้นแบ่ง.....	8
ภาพประกอบที่ 11 Kernal Function .....	8

## บทที่ 1

### บทนำ

#### 1.1 หลักการและเหตุผล

#### 1.2 วัตถุประสงค์ของโครงการ

#### 1.3 ขอบเขตของโครงการ

1. ข้อมูลที่ใช้ในโครงการเป็นข้อมูลกลุ่มตัวอย่างจำนวน 6965 ระเบียบ ประกอบด้วย ตัวแปรทั้งหมด 31 แอตทริบิวต์ แบ่งเป็นตัวแปรต้น 30 แอตทริบิวต์ และตัวแปรตาม 1 แอตทริบิวต์

- Savanna fires: การปล่อยก๊าซจากไฟในระบบนิเวศทุ่งหญ้าสะวันนา
- Forest fires: การปล่อยก๊าซจากไฟในพื้นที่ป่า
- Crop Residues: การปล่อยก๊าซจากการเผาหรือการย่อยสลายเศษพืชหลังการเก็บเกี่ยว
- Rice Cultivation: การปล่อยก๊าซมีเทนจากการปลูกข้าว
- Drained organic soils (CO<sub>2</sub>): การปล่อยก๊าซคาร์บอนไดออกไซด์เมื่อระบายน้ำในดินอินทรีย์
- Pesticides Manufacturing: การปล่อยก๊าซจากการผลิตสารกำจัดศัตรูพืช
- Food Transport: การปล่อยก๊าซจากการขนส่งสินค้าอาหาร
- Forestland: ที่ดินที่ปกคลุมด้วยป่าไม้
- Net Forest conversion: การเปลี่ยนแปลงพื้นที่ป่าเนื่องจากการตัดไม้ทำลายป่าและการปลูกป่า
- Food Household Consumption: การปล่อยก๊าซจากการบริโภคอาหารที่ระดับครัวเรือน
- Food Retail: การปล่อยก๊าซจากการดำเนินงานของร้านค้าปลีกที่ขายอาหาร
- On-farm Electricity Use: การใช้ไฟฟ้าในฟาร์ม
- Food Packaging: การปล่อยก๊าซจากการผลิตและการกำจัดบรรจุภัณฑ์อาหาร
- Agrifood Systems Waste Disposal: การปล่อยก๊าซจากการกำจัดขยะในระบบอาหารและเกษตร
- Food Processing: การปล่อยก๊าซจากการแปรรูปผลิตภัณฑ์อาหาร
- Fertilizers Manufacturing: การปล่อยก๊าซจากการผลิตปุ๋ย
- IPPU: การปล่อยก๊าซจากกระบวนการอุตสาหกรรมและการใช้ผลิตภัณฑ์

- Manure applied to Soils: การปล่อยก๊าซจากการใช้ปุ๋ยคอกในดินเกษตร
- Manure left on Pasture: การปล่อยก๊าซจากปุ๋ยคอกในทุ่งหญ้าหรือพื้นที่เลี้ยงสัตว์
- Manure Management: การปล่อยก๊าซจากการจัดการและการบำบัดปุ๋ยคอก
- Fires in organic soils: การปล่อยก๊าซจากไฟในดินอินทรีย์
- Fires in humid tropical forests: การปล่อยก๊าซจากไฟในป่าฝนเขตร้อนชื้น
- On-farm energy use: การใช้พลังงานในฟาร์ม
- Rural population: จำนวนคนที่อาศัยอยู่ในพื้นที่ชนบท
- Urban population: จำนวนคนที่อาศัยอยู่ในพื้นที่เมือง
- Total Population - Male: จำนวนประชากรชายทั้งหมด
- Total Population - Female: จำนวนประชากรหญิงทั้งหมด
- total\_emission: การปล่อยก๊าซเรือนกระจกทั้งหมดจากแหล่งต่าง ๆ
- Average Temperature °C: อุณหภูมิเฉลี่ยที่เพิ่มขึ้น (ตามปี) ในองศาเซลเซียส

2. ข้อมูลถูกเก็บรวบรวมในรูปแบบไฟล์ CSV และสามารถดาวน์โหลดได้ที่

[Agri-food CO2 emission dataset - Forecasting ML | Kaggle](#)

3. แหล่งที่มาของข้อมูลได้รับข้อมูลมาจากองค์การอาหารและการเกษตร (FAO) และข้อมูลจาก IPCC

4. ทำการทำการทำนายการปล่อยคาร์บอน

5. ทำการทำการทำนายการปล่อยคาร์บอน เพื่อหาแบบจำลองที่ดีที่สุดโดยใช้อัลกอริทึม ซึ่งประกอบไปด้วย

Decision Tree, Random Forest, Linear Regressions, SVM และ Vote Ensembles

6. ทำการศึกษาปัจจัยที่เกี่ยวข้องกับการปล่อยคาร์บอน โดยใช้เทคนิคเหมือนข้อมูลและเทคนิคแบบร่วมกันตัดสินใจ

7. ทำการวัดประสิทธิภาพในการปล่อยคาร์บอน โดยใช้วิธี Mean Absolute Error (ค่าคลาดเคลื่อนกำลังสองเฉลี่ย), Root Mean Squared Error (ค่าคลาดเคลื่อนกำลังสองเฉลี่ย)

8. ทำการแบ่งข้อมูลแบบ 10-Fold Cross validation

9. นำอัลกอริทึมทั้ง 4 มาทำเป็น Ensemble แบบ Voting

10. เครื่องมือที่ใช้ในการทำโครงการ Google Colab

11. การนำเสนอผลการวัดประสิทธิภาพเป็นรูปแบบกราฟ และ Dashboard

## 1.4 ขั้นตอนการดำเนินงาน

ขั้นตอนการดำเนินงานมีทั้งหมด 6 ขั้นตอน

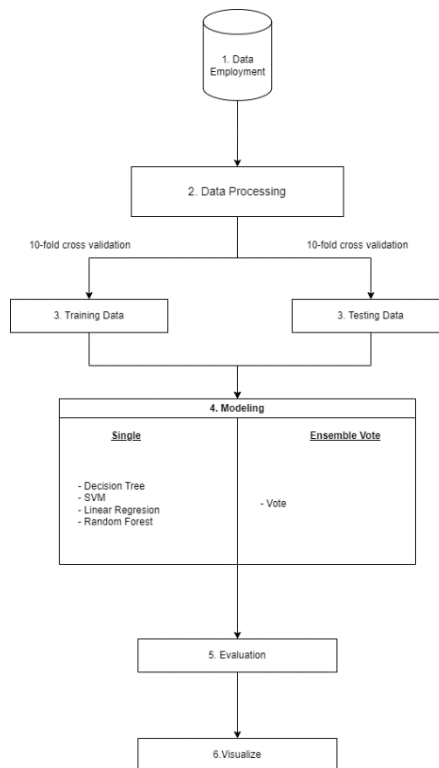
1. เริ่มต้นจากการรวบรวมข้อมูลจาก Kaggle
2. เตรียมข้อมูลโดยการทำความสะอาดข้อมูล
3. แบ่งข้อมูล แบบ Training data และ Testing data ด้วย 10-Fold Cross validation
4. สร้างแบบจำลองโดยใช้อัลกอริทึม Decision Tree, Random Forest, SVM และ Linear

Regressions โดยใช้ Google Colab ทำการวัดประสิทธิภาพและนำอัลกอริทึมทั้ง 4 มาทำ Ensemble แบบ Voting เพื่อสรุปว่าอัลกอริทึมส่วนใหญ่ทำนายแบบใด

5. วิเคราะห์ผลการวัดประสิทธิภาพของการจำแนก

6. จากนั้นจะได้แบบจำลองที่ดีที่สุดแล้วนำมาแสดงผลการทดลองด้วย Visualization

ดังภาพประกอบที่ 1.1



ภาพประกอบที่ 1 แผนผังขั้นตอนการดำเนินงาน

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

สามารถศึกษาการเปรียบเทียบการปล่อยคาร์บอน เพื่อช่วยเฝ้าระวังและเตรียมความพร้อมกับการรับมือกับสภาพอากาศที่ร้อนจัดหรือมลพิษทางอากาศ เช่น การเดินทางหรือทำกิจกรรมกลางแจ้ง

## 1.6 อุปกรณ์และเครื่องมือที่ใช้ในการดำเนินงาน

ฮาร์ดแวร์ อุปกรณ์ที่เป็นสิ่งสำคัญและต้องนำมาใช้ในการทำโปรเจค ประกอบด้วย

คอมพิวเตอร์เครื่องที่ 1 หน่วยประมวลผลกลาง (CPU) รุ่น Intel(R) Core(TM) i5-1135G7

ความเร็ว 2.40 GHz, 2.42 GHz

หน่วยความจำหลัก Installed RAM 8.00 GB

ระบบปฏิบัติการ Windows 11 Home Single Language

คอมพิวเตอร์เครื่องที่ 2 หน่วยประมวลผลกลาง (CPU ) AMD Ryzen 5 4600H

ความเร็ว 3.00 GHz

หน่วยความจำหลัก Installed RAM 16.0 GB

## ระบบปฏิบัติการ Windows 11 Home Single Language

## 2.ซอฟต์แวร์ที่ใช้ Google Colab ในการสร้างอัลกอริทึมและสร้างกราฟแสดงผลข้อมูล

## 1.7 แผนการดำเนินงาน

โครงการปริญญานิพนธ์ฉบับนี้ ดำเนินงาน ณ คณะวิทยาการสารสนเทศ มหาวิทยาลัย มหาสารคามระหว่าง  
เดือน มิถุนายน 2567 ถึง พฤษภาคม 2567 ดังตารางที่ 1.1

## ตารางที่ 1.1 แผนการดำเนินงาน

กิจกรรม	เดือน											
	.ค.	.พ.	.ค.	ม.ย.	.ค.	.ย.	.ค.	.ค.	.ย.	.ค.	.ย.	.ค.
1. ศึกษา และรวบรวมข้อมูล												
2. วิเคราะห์และกำหนด ขอบเขต												
3. ออกแบบขั้นตอนการ ดำเนินการ												

4. พัฒนา โครงการ												
5. ทดสอบและวัด ประสิทธิภาพ												
6. ทำ รายงานสรุป												
7. นำเสนอโครงการ												

## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

## บทที่ 3

### วิธีดำเนินโครงการ

#### 3.1 แผนการดำเนินงาน

#### 3.2 การเตรียมข้อมูล

ศึกษาปัจจัยต่าง ๆ ที่สามารถปล่อยคาร์บอนสู่ชั้นบรรยากาศหรือก่อให้เกิดก๊าซเรือนกระจก ได้แก่ การปล่อยก๊าซจากไฟในระบบนิเวศทุ่งหญ้าสะวันนา, การปล่อยก๊าซจากไฟในพื้นที่ป่า, การปล่อยก๊าซจากการเผาหรือการย่อยสลายเศษพืชหลังการเก็บเกี่ยว, การปล่อยก๊าซมีเทนจากการปลูกข้าว, การปล่อยก๊าซ,คาร์บอนไดออกไซด์เมื่อระบายน้ำในดินอินทรีย์, การปล่อยก๊าซจากการผลิตสารกำจัดศัตรูพืช,การปล่อยก๊าซจากการขนส่งสินค้าอาหาร เป็นต้น เพื่อนำข้อมูลดังกล่าวมาทำเหมืองข้อมูล โดยใช้ Python ในการสร้าง Model เพื่อนำมาวิเคราะห์ซึ่งข้อมูลนี้จัดอยู่ในรูปแบบ CSV เป็นข้อมูลกลุ่มตัวอย่างจำนวน 6965 ระเบียบ ประกอบด้วย ตัวแปรทั้งหมด 31 แอตทริบิวต์ แบ่งเป็นตัวแปรต้น 30 แอตทริบิวต์ และตัวแปรตาม 1 แอตทริบิวต์ โดยมีปัจจัยที่สามารถนำมาวิเคราะห์ได้ ดังตารางที่ 1

ตารางที่ 1

Area	Year	Savanna fires	Forest fires	Crop Residues
Afghanistan	1990	14.7237	0.0557	205.6077
Afghanistan	1991	14.7237	0.0557	209.4971
Afghanistan	1992	14.7237	0.0557	196.5341
Afghanistan	1993	14.7237	0.0557	230.8175



ดูค่าสถิติ (Statistics) เพื่อตรวจสอบข้อมูลอีกครั้ง ว่าแต่ละAttribute มีลักษณะข้อมูลเป็นแบบ ไหน มีค่ามากที่สุด น้อยสุด และค่าเฉลี่ย อยู่ที่ไหน จากตารางจะเห็นว่า มี Attribute ชื่อ Savanna fires, Forest fires ,Crop Residues มีค่า Missing จึงจำเป็นต้องกำจัดออกด้วยการทำความสะอาดข้อมูล ดังตารางที่ 2

ตารางที่ 2

Area	0
Year	0
Savanna fires	31
Forest fires	93
Crop Residues	1389

### 3.3 การทำความสะอาดข้อมูล

ข้อมูลที่ได้นำมาตรวจสอบนั้นมีบางแอตทริบิวต์ที่มีค่าที่ผิดปกติที่ไม่มีข้อมูลที่จะสามารถนำมาประมวลผลได้ เป็นต้น ซึ่งค่าดังกล่าวอาจจะเกิดจากความผิดพลาดจากการเก็บรวบรวมข้อมูล ดังนั้น โครงการนี้จึงทำการตรวจสอบข้อมูลที่สูญหายไปด้วยคำสั่ง fillna จะทำการ เพิ่มค่าข้อมูลที่สูญหายไปด้วยค่าเฉลี่ย ภาพประกอบที่ 14 การตรวจสอบข้อมูลที่สูญหาย

```
df["Savanna fires"].fillna(df["Savanna fires"].mean(), inplace = True)
df["Forest fires"].fillna(df["Forest fires"].mean(), inplace = True)
df["Crop Residues"].fillna(df["Crop Residues"].mean(), inplace = True)
df["Forestland"].fillna(df["Forestland"].mean(), inplace = True)
df["Net Forest conversion"].fillna(df["Net Forest conversion"].mean(), inplace = True)
df["Food Household Consumption"].fillna(df["Food Household Consumption"].mean(), inplace = True)
df["IPPU"].fillna(df["IPPU"].mean(), inplace = True)
df["Manure applied to Soils"].fillna(df["Manure applied to Soils"].mean(), inplace = True)
df["Manure Management"].fillna(df["Manure Management"].mean(), inplace = True)
df["Fires in humid tropical forests"].fillna(df["Fires in humid tropical forests"].mean(), inplace = True)
df["On-farm energy use"].fillna(df["On-farm energy use"].mean(), inplace = True)
```

ภาพประกอบที่ 14 การตรวจสอบข้อมูลที่สูญหาย

### 3.4 การแปลงข้อมูล

ทำการตรวจสอบข้อมูลเพื่อหา Attribute ที่จะนำไปเป็นตัวแปรทำนายผลและทำการแปลงข้อมูล Attribute นั้นซึ่งเป็นตัวหนังสือ ที่เป็นข้อความ (categorical) ให้กลายเป็นตัวเลขจากนั้นทำการเลือก Label ซึ่งเป็นตัวแปรตามมาเป็นค่าที่ใช้ทำนายได้ ดังภาพประกอบที่ 15 และ 16

#	Column	Non-Null Count	Dtype
0	Area	6965 non-null	object
1	Year	6965 non-null	int64
2	Savanna fires	6934 non-null	float64

ภาพประกอบที่ 15 Attribute ที่จะทำการแปลง

...			
29	total_emission	6965 non-null	float64
30	Average Temperature °C	6965 non-null	float64
(dtype: float64(29), int64(1), object(1))			

ภาพประกอบที่ 16 Attribute ที่จะใช้เป็น Label

	Area	Year	Savanna fires	Forest fires	Crop Residues	Rice Cultivation	D
0	Afghanistan	1990	14.7237	0.0557	205.6077	686.00	
1	Afghanistan	1991	14.7237	0.0557	209.4971	678.16	
2	Afghanistan	1992	14.7237	0.0557	196.5341	686.00	
3	Afghanistan	1993	14.7237	0.0557	230.8175	686.00	
4	Afghanistan	1994	14.7237	0.0557	242.0494	705.60	
5	Afghanistan	1995	14.7237	0.0557	243.8152	666.40	
6	Afghanistan	1996	38.9302	0.2014	249.0364	686.00	
7	Afghanistan	1997	30.9378	0.1193	276.2940	705.60	
8	Afghanistan	1998	64.1411	0.3263	287.4346	705.60	
9	Afghanistan	1999	46.1683	0.0895	247.4980	548.80	
0 rows x 31 columns							

ภาพประกอบที่ 17 ตัวอย่างตารางก่อนแปลงข้อมูล

	Area	Year	Savanna fires	Forest fires	Crop Residues	Rice Cultivation
0	0	1990	14.7237	0.0557	205.6077	686.00
1	0	1991	14.7237	0.0557	209.4971	678.16
2	0	1992	14.7237	0.0557	196.5341	686.00
3	0	1993	14.7237	0.0557	230.8175	686.00
4	0	1994	14.7237	0.0557	242.0494	705.60
5	0	1995	14.7237	0.0557	243.8152	666.40
6	0	1996	38.9302	0.2014	249.0364	686.00
7	0	1997	30.9378	0.1193	276.2940	705.60
8	0	1998	64.1411	0.3263	287.4346	705.60
9	0	1999	46.1683	0.0895	247.4980	548.80

ภาพประกอบที่ 18 ตัวอย่างตารางหลังแปลงข้อมูล

### 3.5 Backward-Elimination

Backward Elimination เป็นเทคนิคหนึ่งในการเลือกตัวแปร (feature selection) ที่ใช้ในการสร้างโมเดลเชิงสถิติ โดยเฉพาะอย่างยิ่งในการสร้างโมเดลการถดถอย (regression model) กระบวนการนี้ช่วยลดจำนวนตัวแปรที่ไม่จำเป็นหรือไม่มีความสำคัญทางสถิติ ซึ่งจะช่วยให้โมเดลของมีความเรียบง่ายและทำงานได้มีประสิทธิภาพมากขึ้น

```

y = df['Average Temperature °C']
X = df.drop(['Average Temperature °C'], axis=1)

X = sm.add_constant(X)

model = sm.OLS(y, X).fit()
print(model.summary())

pmax = 1
while (len(X.columns) > 0):
    pmax = max(model.pvalues)
    if pmax > 0.05:
        feature_with_p_max = model.pvalues.idxmax()
        X = X.drop([feature_with_p_max], axis=1)
        model = sm.OLS(y, X).fit()
    else:
        break

selected_features = X.columns
selected_features = selected_features.drop('const')
print(selected_features)

```

ภาพประกอบที่ 19 การทำ backward-elimination โดยใช้ python

OLS Regression Results							
=====							
Dep. Variable:	Average Temperature °C	R-squared:	0.329				
Model:	OLS	Adj. R-squared:	0.326				
Method:	Least Squares	F-statistic:	113.5				
Date:	Mon, 05 Aug 2024	Prob (F-statistic):	0.00				
Time:	01:03:46	Log-Likelihood:	-4401.9				
No. Observations:	6965	AIC:	8866.				
Df Residuals:	6934	BIC:	9078.				
Df Model:	30						
Covariance Type:	nonrobust						
=====							
		coef	std err	t	P> t	[0.025	0.975]
-----							
const		-69.4626	1.273	-54.568	0.000	-71.958	-66.967
Area		-0.0002	8.34e-05	-2.407	0.016	-0.000	-3.73e-05
Year		0.0351	0.001	55.277	0.000	0.034	0.036
Savanna fires		-4.933e-06	1.69e-06	-2.925	0.003	-8.24e-06	-1.63e-06
Forest fires		-1.535e-05	5.18e-06	-2.962	0.003	-2.55e-05	-5.19e-06
Crop Residues		-4.344e-05	1.03e-05	-4.228	0.000	-6.36e-05	-2.33e-05
Rice Cultivation		-6.414e-06	1.41e-06	-4.564	0.000	-9.17e-06	-3.66e-06
Drained organic soils (CO2)		-3.352e-06	1.08e-06	-3.100	0.002	-5.47e-06	-1.23e-06
Pesticides Manufacturing		-2.289e-05	1.22e-05	-1.877	0.061	-4.68e-05	1.02e-06
Food Transport		8.041e-06	3.72e-06	2.161	0.031	7.45e-07	1.53e-05
Forestland		-4.304e-06	8.47e-07	-5.081	0.000	-5.97e-06	-2.64e-06
Net Forest conversion		-3.931e-06	8.59e-07	-4.579	0.000	-5.61e-06	-2.25e-06
Food Household Consumption		-2.998e-06	1.9e-06	-1.581	0.114	-6.71e-06	7.2e-07
Food Retail		-1.27e-05	3.31e-06	-3.836	0.000	-1.92e-05	-6.21e-06
On-farm Electricity Use		2.554e-06	2.51e-06	1.016	0.310	-2.37e-06	7.48e-06
Food Packaging		-1.625e-06	4.56e-06	-0.356	0.722	-1.06e-05	7.31e-06
AgriFood Systems Waste Disposal		-4.395e-07	2.15e-06	-0.204	0.838	-4.66e-06	3.78e-06
Food Processing		-3.908e-06	1.74e-06	-2.250	0.024	-7.31e-06	-5.04e-07
Fertilizers Manufacturing		-1.323e-05	2.51e-06	-5.280	0.000	-1.81e-05	-8.32e-06
IPPU		-2.559e-06	8.28e-07	-3.092	0.002	-4.18e-06	-9.37e-07
Manure applied to Soils		7.83e-05	2.07e-05	3.790	0.000	3.78e-05	0.000
Manure left on Pasture		-1.133e-05	2.05e-06	-5.525	0.000	-1.54e-05	-7.31e-06
Manure Management		-4.976e-07	8.25e-06	-0.060	0.952	-1.67e-05	1.57e-05
Fires in organic soils		-4.686e-06	9.02e-07	-5.193	0.000	-6.46e-06	-2.92e-06
Fires in humid tropical forests		5.222e-06	5.77e-06	0.904	0.366	-6.1e-06	1.65e-05
On-farm energy use		-3.514e-06	1.98e-06	-1.774	0.076	-7.4e-06	3.7e-07
Rural population		-2.093e-09	2.09e-09	-0.999	0.318	-6.2e-09	2.01e-09
Urban population		-5.995e-09	2.43e-09	-2.469	0.014	-1.08e-08	-1.24e-09
Total Population - Male		4.771e-09	6.9e-09	0.691	0.490	-8.76e-09	1.83e-08
Total Population - Female		-4.963e-10	7.64e-09	-0.065	0.948	-1.55e-08	1.45e-08
total_emission		3.964e-06	8.58e-07	4.618	0.000	2.28e-06	5.65e-06
=====							
Omnibus:	243.770	Durbin-Watson:	1.252				
Prob(Omnibus):	0.000	Jarque-Bera (JB):	493.747				
Skew:	0.245	Prob(JB):	6.08e-108				
Kurtosis:	4.209	Cond. No.	3.58e+10				
=====							
Notes:							
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.							
[2] The condition number is large, 3.58e+10. This might indicate that there are strong multicollinearity or other numerical problems.							
Index(['Area', 'Year', 'Savanna fires', 'Forest fires', 'Crop Residues', 'Rice Cultivation', 'Drained organic soils (CO2)', 'Pesticides Manufacturing', 'Food Transport', 'Forestland', 'Net Forest conversion', 'Food Household Consumption', 'Food Retail', 'Food Processing', 'Fertilizers Manufacturing', 'IPPU', 'Manure applied to Soils', 'Manure left on Pasture', 'Fires in organic soils', 'On-farm energy use', 'Urban population', 'total_emission'], dtype='object')							

ภาพประกอบที่ 20 ผลลัพธ์ของการทำ backward-elimination

```
selected_features
```

```
Index(['Area', 'Year', 'Savanna fires', 'Forest fires', 'Crop Residues',  
      'Rice Cultivation', 'Drained organic soils (CO2)',  
      'Pesticides Manufacturing', 'Food Transport', 'Forestland',  
      'Net Forest conversion', 'Food Household Consumption', 'Food Retail',  
      'Food Processing', 'Fertilizers Manufacturing', 'IPPU',  
      'Manure applied to Soils', 'Manure left on Pasture',  
      'Fires in organic soils', 'On-farm energy use', 'Urban population',  
      'total_emission'],  
      dtype='object')
```

ภาพประกอบที่ 21 features จากการทำ backward-elimination

### 3.6 Cross validation

Cross validation คือการ แบ่งข้อมูลไป Train ในงานวิจัยนี้ได้ Setting ค่าแบ่งข้อมูลเป็น 10 folds เพื่อแบ่งข้อมูลใช้สำหรับ Train และ Test โดยสลับให้ข้อมูลแต่ละส่วนนำมาเป็นชุด Test เพื่อทดสอบประสิทธิภาพ โดยงานวิจัยนี้จะทำการทดสอบทั้งสี่โมเดล คือ Decision Tree, Random Forest, SVM และ Linear Regression ทั้งหมด 10 ครั้งเพื่อวัดประสิทธิภาพของทั้งสี่โมเดล

```
dt = DecisionTreeRegressor()  
rf = RandomForestRegressor()  
lr = LinearRegression()  
svr = SVR()  
voting = VotingRegressor([('dt', dt), ('rf', rf), ('lr', lr), ('svr', svr)])  
  
kf = KFold(n_splits=10, shuffle=True, random_state=42)
```

ภาพประกอบที่ 22 Cross validation

### 3.7 การสร้าง Model ในการจำแนก

จะใช้ IDE ของ VS Code เพื่อใช้สร้างตัวแบบในการวิเคราะห์ข้อมูลเพื่อทำนายการปล่อยคาร์บอน โดยใช้เทคนิค Decision Tree, Random Forest, SVM, Linear Regression และ Vote Ensemble แล้วนำผลพยากรณ์มาเปรียบเทียบกับค่าความถูกต้องโดยใช้ผลที่มีความเชื่อถือมากที่สุด

#### 3.7.1.1 ขั้นตอนการสร้าง Model Decision tree, Random Forest, SVM และ Linear regression

เป็นเทคนิคที่นำข้อมูลแต่ละโนด (Node) ของแอททริบิวต์ (Attribute) มาทำการตัดสินใจ จากนั้นจะแสดงข้อมูลออกมาเป็นกิ่ง (Branch) และแสดงค่าออกมาเป็นใบ (Leaf) โดยใช้ information Gain มาหาความสัมพันธ์ในแต่ละโนดและทำให้ต้นไม้การตัดสินใจมีความซับซ้อนไม่มาก การสร้างโมเดล Decision Tree, , Random Forest, SVM และ Linear regression โดยใช้ Python นั้นทำได้โดยใช้ไลบรารี scikit-learn (sklearn) ซึ่งเป็นไลบรารีที่นิยมใช้ในงาน Machine Learning ในที่นี้จะอธิบายขั้นตอนตั้งแต่การเตรียมข้อมูล การสร้างโมเดล การฝึกสอนโมเดล ไปจนถึงการประเมินผล

##### 3.7.1.1.1 เตรียมข้อมูล

ในขั้นตอนนี้เราจะโหลดข้อมูลที่ใช้ในการฝึกสอนโมเดล โดยที่ X คือการนำ feature ที่ผ่านการทำ Backward-elimination มาแล้ว และ y คือ Label ที่จะใช้ในการทำนาย

```
x = df[selected_features]
y = df['Average Temperature °C']
```

##### 3.7.1.1.2 สร้างและฝึกสอนโมเดล Decision Tree, Random Forest, SVM และ Linear regression

ในขั้นตอนนี้เราจะสร้างโมเดล และฝึกสอนโมเดลด้วยข้อมูลที่แบ่งไว้โดยใช้ K-fold โดยที่

n\_splits=10 คือการแบ่งข้อมูลเป็น 10 ส่วน (หรือ 10 folds) ซึ่งหมายความว่าแต่ละครั้งของการ cross-validation จะมี 1 fold ที่ใช้สำหรับการทดสอบ และอีก 9 fold ที่เหลือใช้สำหรับการฝึกสอน (training) กระบวนการนี้จะถูกทำซ้ำทั้งหมด 10 ครั้ง โดยแต่ละครั้งจะใช้ fold ที่ต่างกันในการทดสอบ จนครบทุก fold

shuffle=True คือการตั้งค่า shuffle=True หมายความว่าข้อมูลจะถูกสุ่มก่อนที่จะแบ่งเป็น folds การสุ่มข้อมูลช่วยให้แน่ใจว่าข้อมูลแต่ละ fold มีความหลากหลายและไม่เกิดการแบ่งข้อมูลที่ไม่สมดุล เช่น การที่ข้อมูลที่อยู่ติดกันมีลักษณะคล้ายกันมากเกินไป

random\_state=42 ใช้ในการตั้ง seed สำหรับการสุ่มหมายเลข ซึ่งทำให้การสุ่มข้อมูลเป็นแบบ deterministic (สามารถทำซ้ำได้) หมายเลข seed สามารถเป็นค่าตัวเลขใดก็ได้ เช่น 42 ในที่นี้เป็นค่า seed ที่มักใช้เพื่อให้การสุ่มเป็นแบบ reproducible (สามารถได้ผลลัพธ์เดิมเมื่อทำซ้ำ) การใช้ random\_state จะช่วยให้ทุกครั้งที่เรารันคำสั่งนี้ ผลลัพธ์ของการสุ่มจะเหมือนเดิม ทำให้การทดสอบโมเดลมีความเสถียร

```
kf = KFold(n_splits=10, shuffle=True, random_state=42)
```

### 3.7.1.2 ทำนายผลและประเมินผลโมเดล

หลังจากฝึกสอนโมเดลแล้ว เราจะใช้ข้อมูลชุดทดสอบในการทำนายผลและประเมินประสิทธิภาพของโมเดล

#### 3.7.1.2.1 การทำ Cross-Validation สำหรับ MSE

```
mse_scores = cross_val_score(model, X, y, cv=kf, scoring=mse_scorer)
```

โดยที่ cross\_val\_score: ฟังก์ชันนี้ทำ cross-validation สำหรับโมเดลที่กำหนด

model: โมเดลปัจจุบันในรูป, X: ฟีเจอร์ของข้อมูล, y: เป้าหมาย (target), cv=kf: ใช้การแบ่งข้อมูลแบบ KFold ที่เรากำหนดไว้ก่อนหน้านี้, scoring=mse\_scorer: ใช้ MSE (Mean Squared Error) เป็นตัวชี้วัดประสิทธิภาพ

#### 3.7.1.2.2 คำนวณ RMSE

```
rmse_scores = np.sqrt(mse_scores)
```

 โดยใช้ np.sqrt เพื่อคำนวณรากที่สองของค่า MSE สำหรับแต่ละ fold ซึ่งจะได้ค่า RMSE (Root Mean Squared Error)

#### 3.7.1.2.3 การทำ Cross-Validation สำหรับ MAE

```
mae_scores = cross_val_score(model, X, y, cv=kf, scoring=mae_scorer)
```

โดย ใช้การทำ cross-validation เช่นเดียวกับ MSE แต่เปลี่ยนตัวชี้วัดเป็น MAE

(Mean Absolute Error)

#### 3.7.1.2.4 การทำ Cross-Validation สำหรับ $R^2$ :

```
r2_scores = cross_val_score(model, X, y, cv=kf, scoring=r2_scorer)
```

โดย ใช้การทำ cross-validation เช่นเดียวกับ MSE แต่เปลี่ยนตัวชี้วัดเป็น  $R^2$  (R-squared score)

#### 3.7.1.2.5 คำนวณค่าเฉลี่ยและพิมพ์ผลลัพธ์:

model.\_\_class\_\_.\_\_name\_\_: ชื่อของคลาสโมเดล เช่น DecisionTreeRegressor

np.mean(mse\_scores): ค่าเฉลี่ยของ MSE จากการ cross-validation

np.mean(rmse\_scores): ค่าเฉลี่ยของ RMSE จากการคำนวณค่า RMSE

np.mean(mae\_scores): ค่าเฉลี่ยของ MAE จากการ cross-validation

np.mean(r2\_scores): ค่าเฉลี่ยของ  $R^2$  จากการ cross-validation

```
models = [dt, rf, lr, svr, voting]
for model in models:
    mse_scores = cross_val_score(model, X, y, cv=kf, scoring=mse_scorer)
    rmse_scores = np.sqrt(mse_scores)
    mae_scores = cross_val_score(model, X, y, cv=kf, scoring=mae_scorer)
    r2_scores = cross_val_score(model, X, y, cv=kf, scoring=r2_scorer)

    print(f'{model.__class__.__name__} Mean MSE: {np.mean(mse_scores)}, Mean RMSE: \
|   {np.mean(rmse_scores)}, Mean MAE: {np.mean(mae_scores)}, Mean R^2: {np.mean(r2_scores)}')
    print("\n")
```

✓ 17m 58.4s

ผลลัพธ์ของโมเดล ที่ใช้สำหรับการทำนายโดยมีค่า MSE, RMSE, MAE และ  $R^2$  ดังตารางที่ 3



ตารางที่ 3

Model	การวัดประสิทธิภาพ			
	MSE	RMSE	MAE	R <sup>2</sup>
Decision Tree	0.2295	0.4788	0.3462	0.2414
Random Forest	0.1265	0.3554	0.2576	0.5900
Linear Regression	0.2090	0.4569	0.3447	0.3220
SVM	0.3058	0.5527	0.4236	0.0084

### 3.7.2 เทคนิคโหวตร่วม (Vote Ensemble)

เทคนิคโหวตร่วม (Vote Ensemble) เป็นโมเดลที่ฝึกกับโมเดลต่างๆและคาดการณ์ เอาต์พุตตามความน่าจะเป็นสูงสุดของคลาสที่เลือกเป็นเอาต์พุต โดยรวบรวมผลการค้นพบของแต่ละ เทคนิค (เทคนิคต้นไม้การตัดสินใจ เทคนิคนาอ็ฟ และเทคนิคป่าสุ่ม) ที่ส่งผ่านไปยังตัวจำแนกการ ลงคะแนนเสียง (Voting Regressor) และคาดการณ์ระดับผลลัพธ์ตามเสียงส่วนใหญ่โหวตสูงสุด

การวัดประสิทธิภาพ ด้วย Confusion matrix ทั้งหมด 4 การวัดประสิทธิภาพ คือ MSE, RMSE, MAE, R<sup>2</sup> ผลลัพธ์ของโมเดล Vote ที่ใช้สำหรับการทำนาย โดยมีค่า MSE, RMSE, MAE, R<sup>2</sup> ดังตารางที่ 4

ตารางที่ 4

Model	การวัดประสิทธิภาพ			
	MSE	RMSE	MAE	R <sup>2</sup>
Voting	0.1550	0.3934	0.2911	0.4973

### 3.8 สมการที่ใช้ในการวัดประสิทธิภาพ

1. MAE

$$MAE = 1/n * \sum |prediction - actual|$$

2. MSE

$$MSE = 1/n * \sum (prediction - actual)^2$$

3. RMSE

$$RMSE = \sqrt{1/n * \sum (prediction - actual)^2}$$

4. R2

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

## เอกสารอ้างอิง

- [1] Cortez P, Morais AdjR. A data mining approach to predict forest fires using meteorological data.
- [2] Chalathip. รายงานฟาด 57 บริษัทตัวการโลกร้อน ปล่อยก๊าซคาร์บอนฯรวม 80%. 28]; <https://workpointtoday.com/sustainability-climate-change-co2-emissions/>.
- [3] ทศนนาคะจิตต์ ภ. คาร์บอนไดออกไซด์ไม่มีประโยชน์เลยหรือ. 28]; <https://www.scimath.org/article-physics/item/9827-2019-02-21-08-51-20>.
- [4] Wikipedia. คาร์บอนไดออกไซด์. 28]; <https://th.wikipedia.org/wiki/%E0%B8%84%E0%B8%B2%E0%B8%A3%E0%B9%8C%E0%B8%9A%E0%B8%AD%E0%B8%99%E0%B9%84%E0%B8%94%E0%B8%AD%E0%B8%AD%E0%B8%81%E0%B9%84%E0%B8%8B%E0%B8%94%E0%B9%8C>.
- [5] co2meter. Dangers of CO2: What You Need to Know. 28]; <https://www.co2meter.com/blogs/news/dangers-of-co2-what-you-need-to-know>.
- [6] Google. ยินดีต้อนรับสู่ Colaboratory. 29]; <https://colab.research.google.com/>.
- [7] amazon. การทำเหมืองข้อมูลคืออะไร. 30]; <https://aws.amazon.com/th/what-is/data-mining/>.
- [8] Wikipedia. ต้นไม้ตัดสินใจ. 29]; <https://th.wikipedia.org/wiki/%E0%B8%95%E0%B9%89%E0%B8%99%E0%B9%84%E0%B8%A1%E0%B9%89%E0%B8%95%E0%B8%B1%E0%B8%94%E0%B8%AA%E0%B8%B4%E0%B8%99%E0%B9%83%E0%B8%88>.
- [9] datacamp. Decision Tree Classification in Python Tutorial. 29]; <https://www.datacamp.com/tutorial/decision-tree-classification-python>.
- [10] BorntoDev. รู้จักกับ Decision Tree มันคือต้นไม้อะไร ทำงานอย่างไร ? 29]; <https://www.borntodev.com/2022/09/15/%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81%E0%B8%81%E0%B8%B1%E0%B8%9A-decision-tree/>.
- [11] Daroontham W. เจาะลึก Random Forest !!!— Part 2 of “รู้จัก Decision Tree, Random Forest, และ XGBoost!!!”. 29]; <https://medium.com/@witchapongdaroontham/%E0%B9%80%E0%B8%88%E0%B8%B2%E0%B8%B0%E0%B8%A5%E0%B8%B6%E0%B8%81-random-forest-part-2-of->

[%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81-decision-tree-random-forest-%E0%B9%81%E0%B8%A5%E0%B8%B0-xgboost-79b9f41a1c1c.](#)

[12] Soontranon N. Random Forest คือ อะไร. 29]; <https://www.nerd-data.com/random-forest/>.

[13] Chomchit P. คู่มือสำหรับ Linear Regression: แนวคิด และ การใช้งาน. 28/7/2024]; <https://medium.com/@phiphatchomchit/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD%E0%B8%AA%E0%B8%B3%E0%B8%AB%E0%B8%A3%E0%B8%B1%E0%B8%9A-linear-regression-%E0%B9%81%E0%B8%99%E0%B8%A7%E0%B8%84%E0%B8%B4%E0%B8%94-%E0%B9%81%E0%B8%A5%E0%B8%B0-%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99-273b10b55472.>

[14] Zheng P. ทำความรู้จัก “Linear Regression” Algorithm ที่คนทำ Machine Learning ยังไงก็ต้องได้ใช้! 30];

<https://www.borntodev.com/2021/08/26/%E0%B8%97%E0%B8%B3%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81-linear-regression-algorithm/>.

[15] เมธาเฉลิมพัฒน์ ร. การประยุกต์ใช้ Machine Learning กับงานในภาคอุตสาหกรรม

[16] hitexts. Model Evaluation มีตัววัดผลอะไรบ้างและทำไปเพื่ออะไร.

<https://www.hitexts.com/data/model-evaluation/>.

[17] Service AW. การแสดงข้อมูลด้วยภาพคืออะไร. 29]; <https://aws.amazon.com/th/what-is/data-visualization/>.

[18] Li S, Tong Z, Haroon M. Estimation of transport CO2 emissions using machine learning algorithm. Transportation Research Part D: Transport and Environment 2024; 133104276.

<https://www.sciencedirect.com/science/article/pii/S1361920924002335>

[19] Kumar S. A novel hybrid machine learning model for prediction of CO2 using socio-economic and energy attributes for climate change monitoring and mitigation policies.

Ecological Informatics 2023; 77102253.

<https://www.sciencedirect.com/science/article/pii/S1574954123002820>