
CHEMICAL OXYGEN DEMAND PREDICTION BASED ON THE SOUTH KOREAN RIVER 'S WATER QUANTITY

KHUNANON SURASORN 6110502634 SECTION I



รายชื่อแม่น้ำในประเทศเกาหลีใต้ที่ใช้ในการศึกษา

แม่น้ำใน
ประเทศ
เกาหลีใต้

แม่น้ำนัคดง (Nakdong River)

แม่น้ำฮัน (Han River)

แม่น้ำยongsan (Yeongsan River)

แม่น้ำกึม (Geum River)

KOREAN STATISTICAL INFORMATION SERVICE

KOSIS

Statistical List : Statistical Dat...

My History

Statistica Search word Search

go top

ASC

- Korea National Environmental Health
- Statistics of Meteorological Observat
- Statistics of Climate Change Monitor
- Status and Trends of Air Quality
- Statistics of Waterworks
- Result of Monitoring on Water Qualit
 - Water Quality of River - Han River (
 - Water Quality of River - Geum Rive
 - Water Quality of River - Nakdong R
 - Water Quality of River - Yeongsan F
 - Water Quality of City (M 2008.01~20
 - Water Quality of Lake (M 2007.01~2
 - Water Quality of Industrial Comple
- Status of Volume Based Waste Fee S
- Major Cities Noise Level
- Result of Soil Quality and Contamina
- Status of Waste Recycling and Recyc
- Statistics of Sewerage
- Seawater Quality
- Status of Investment in Environment
- Energy

1. Water Quality of River ... X CloseAll

Water Quality of River - Nakdong River

Term : Monthly 2007.01 ~ 2020.12 / Data Updated on: 2023-03-22

Time Analysis Pivot Release

New Tab Copy Table URL & Reference Print Download setting

By Water System(1)	By Water System(2)	By Water System(3)	2020.12			
			Biochemical Oxygen Demand(BOD) (mg/ℓ)	Dissolved Oxygen(DO) (mg/ℓ)	Chemical Oxygen Demand(COD) (mg/ℓ)	Water Temperature (Temperature)
Nakdonggang	Andongdam	Hwangji 2	1.6	13.8	3.9	2.3
		Hwangji 3	0.8	13.9	2.8	5.7
		Bonghwa	0.2	14.0	3.9	3.6
		Dosan	0.4	14.5	2.1	1.6
		Andong1	0.5	9.4	6.3	9.7
		Hoeryongcheon	0.3	13.1	1.6	5.0
		Golpocheon	0.3	12.1	1.7	5.5
		Seokpo1	0.5	13.6	3.5	2.4
		Seokpo2	0.5	13.2	3.2	1.4
		Banbyeoncheon1	1.0	12.4	3.1	5.1
		Yongjeoncheon2	0.5	12.8	2.1	5.0
		Yongjeoncheon1	1.1	12.3	2.2	5.0
		Gilancheon1	0.9	13.4	2.6	3.0
		Gilancheon2	0.4	13.8	3.3	2.8
		Banbyeoncheon1A	0.8	13.6	2.7	5.4
	Imhadam	Banbyeoncheon2	0.5	11.7	4.3	7.6
		Banbyeoncheon2-1	0.7	12.7	6.3	5.1

ตัวแปรที่มีในชุดข้อมูล

By water system(1)	By water system(2)	By water system(3)	Biochemical Oxygen Demand(BOD)	Dissolved Oxygen(DO)
Chemical Oxygen Demand(COD)	Water Temperature	Hydrogen ion Concentration	Suspended Solids(SS)	Total Nitrogen(T-N)
Total Phosphorus (T-P)	Fecal Coliform Bacteria	Coliform Bacteria	Electrical Conductivity(EC)	Anionic Surfactant(ABS)
phenol	Cadmium(Cd)	Cyanogen(CN)	Lead(Pb)	Hexavalent chromium Cr+6
Arsenic(As)	Hydrargyrum(Hg)	Nitrate Nitrogen(NO3-N)	Ammonia Nitrogen(NH3-N)	Phosphate-phosphorus(PO4-P)
Chlorophyll a	Dissolved Total Nitrogen(DTN)	Dissolved Total Phosphorus(DTP)	TOC	Antimony
Flow	Organic Phosphorus	Polychlorinated Biphenyl(PCB)	Tetrachloroethylene (PCE)	Depth of water
DEHP				

ตัวแปรหลังผ่านการปรับปรุงชุดข้อมูล

Biochemical Oxygen Demand(BOD)	Dissolved Oxygen(DO)	Chemical Oxygen Demand(COD)	Water Temperature	Hydrogen ion Concentration
Suspended Solids(SS)	Total Nitrogen(T-N)	Total Phosphorus (T-P)	Fecal Coliform Bacteria	Coliform Bacteria
Electrical Conductivity(EC)	Nitrate Nitrogen(NO3-N)	Ammonia Nitrogen(NH3-N)	Phosphate-phosphorus(PO4-P)	Chlorophyll a
Dissolved Total Nitrogen(DTN)	Dissolved Total Phosphorus(DTP)	TOC		

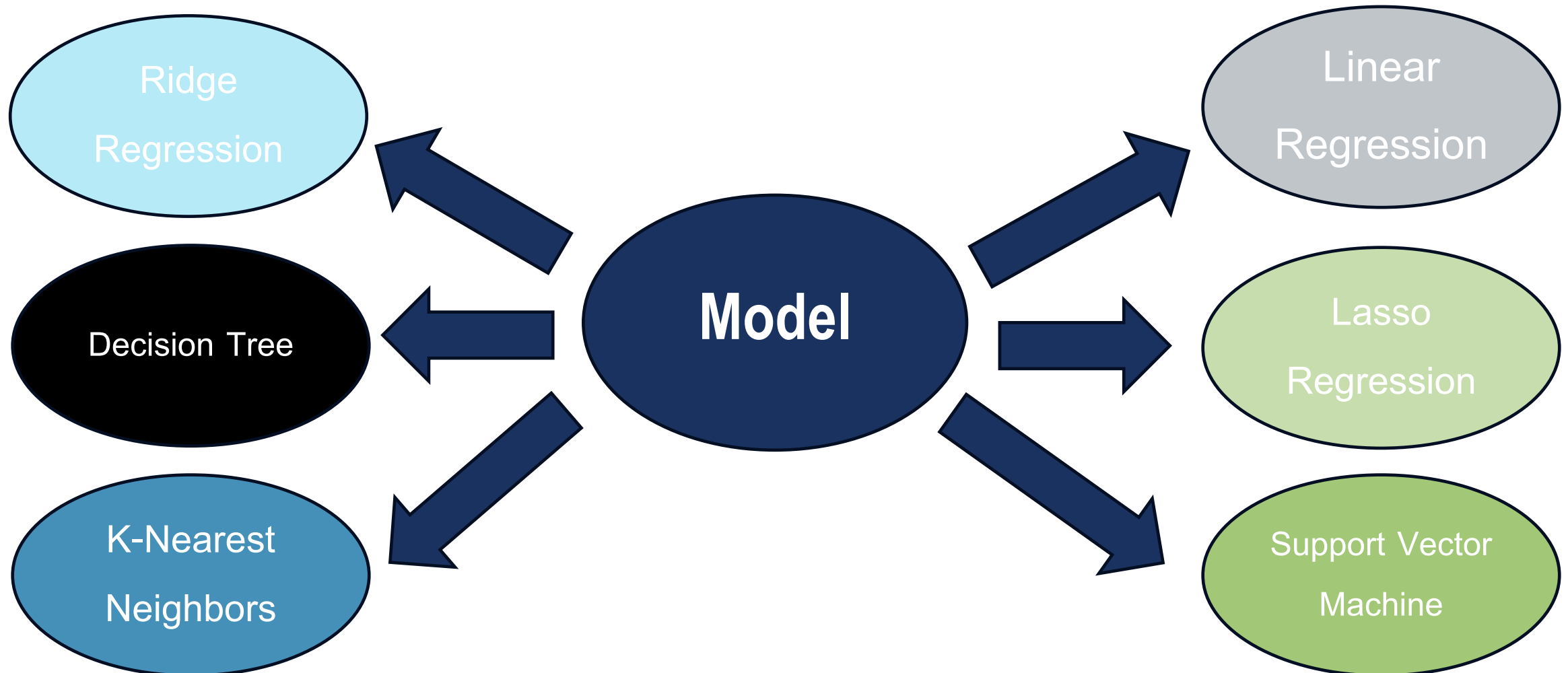
ตัวแปรสำหรับโมเดล

ตัวแปรต้น

Biochemical Oxygen Demand(BOD)	Dissolved Oxygen(DO)	Water Temperature	Hydrogen ion Concentration	Suspended Solids(SS)
Total Nitrogen(T-N)	Total Phosphorus (T-P)	Fecal Coliform Bacteria	Coliform Bacteria	Electrical Conductivity(EC)
Nitrate Nitrogen(NO3-N)	Ammonia Nitrogen(NH3-N)	Phosphate-phosphorus(PO4-P)	Chlorophyll a	Dissolved Total Nitrogen(DTN)
Dissolved Total Phosphorus(DTP)	TOC			

ตัวแปรตาม **Chemical Oxygen Demand(COD)**

โมเดลที่ใช้ในการศึกษา



การประเมินประสิทธิภาพของโมเดลโดยการหาค่าความคลาดเคลื่อน

Mean Absolute Error (MAE)

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}|}{n}$$

Mean Squared Error (MSE)

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y})^2}{n}$$

```
n_m1.fit(x_train,y_train)
ny1_pred = n_m1.predict(x_test)
print("Mean Absolute Error: ",mean_absolute_error(y_test, ny1_pred))
print("Mean Squared Error: ",mean_squared_error(y_test, ny1_pred, squared = True))
```

```
Mean Absolute Error: 0.7987065697497872
Mean Squared Error: 1.217500835963755
```


โมเดลและพารามิเตอร์ที่ได้จากวิธีการ **HYPERPARAMETER TUNING**

โมเดลหลัก (**Main Model**)

```
n_m1 = LinearRegression(fit_intercept = False, n_jobs = None, positive = False)
n_m2 = Ridge(alpha = 10.0, fit_intercept = False, random_state = None)
n_m3 = Lasso(alpha = 0.01, fit_intercept = False, random_state = None)
n_m4 = SVR(C = 10, degree = 3, epsilon = 0.01, gamma = 'auto')
n_m5 = KNeighborsRegressor(algorithm = 'auto', n_neighbors = 7, p = 1, weights = 'distance')
n_m6 = DecisionTreeRegressor(max_depth = None, min_samples_leaf = 4,
                             min_samples_split = 4, random_state = None, splitter = 'random')
```

การประเมินโมเดลหลักโดยข้อมูลทดสอบคุณภาพน้ำของแม่น้ำน่าน (TEST DATA)

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	0.7987	1.2175
Ridge Linear Regression	0.7788	1.1566
Lasso Linear Regression	0.7812	1.1606
Support Vector Regression	1.7267	5.7188
K-Neighbors Regressor	1.5473	4.7114
Decision Tree Regressor	0.8264	1.4836

การประเมินโมเดลหลักโดยข้อมูลคุณภาพน้ำของแม่น้ำยองชั้น

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	1.1237	6.6095
Ridge Linear Regression	1.0395	4.5149
Lasso Linear Regression	1.0515	4.8160
Support Vector Regression	2.5100	11.4438
K-Neighbors Regressor	2.5191	13.0675
Decision Tree Regressor	1.4637	5.0913

การประเมินโมเดลหลักโดยข้อมูลคุณภาพน้ำของแม่น้ำอัน

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	0.9347	1.8513
Ridge Linear Regression	0.9398	1.7709
Lasso Linear Regression	0.9388	1.7764
Support Vector Regression	2.5352	8.6404
K-Neighbors Regressor	2.0270	6.4447
Decision Tree Regressor	1.1488	2.4453

การประเมินโมเดลหลักโดยข้อมูลคุณภาพน้ำของแม่น้ำกิม

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	2.0113	9.9762
Ridge Linear Regression	1.9444	9.8418
Lasso Linear Regression	1.9692	10.1276
Support Vector Regression	2.4173	11.1578
K-Neighbors Regressor	2.6398	13.3086
Decision Tree Regressor	1.8136	6.6792

ผลลัพธ์การประเมินโมเดลของแม่น้ำน้กตงเปรียบเทียบกับชุดข้อมูลของแม่น้ำสายอื่น

Model	Mean Square Error			
	Nakdong River Test Dataset	Yeongsan River Dataset	Han river Dataset	Geum river Dataset
Linear Regression	1.2175	6.6095	1.8513	9.9762
Ridge Linear Regression	1.1566	4.5149	1.7709	9.8418
Lasso Linear Regression	1.1606	4.8160	1.7764	10.1276
Support Vector Regression	5.7188	11.4438	8.6404	11.1578
K-Neighbors Regressor	4.7114	13.0675	6.4447	13.3086
Decision Tree Regressor	1.4836	5.0913	2.4453	6.6792

การปรับปรุงโมเดล

1. เปลี่ยนชุดข้อมูลที่มีข้อมูลคุณภาพน้ำของแม่น้ำสายเดียวเป็นข้อมูลคุณภาพน้ำของแม่น้ำสี่สาย เพื่อประเมินผลโมเดลหลักเปรียบเทียบกัน
2. ใช้คำสั่ง **Hyperparameter Tuning** หาโมเดลด้วย **Mean Absolute Error** อ้างอิงโดยชุดข้อมูลคุณภาพน้ำของแม่น้ำสี่สาย
3. ในการหาโมเดลใหม่จะไม่พิจารณา **Support Vector Regression** และ **K-Neighbors Regressor**

โมเดลและพารามิเตอร์ที่ได้จากวิธีการ **HYPERPARAMETER TUNING**

โมเดลหลัก (**Main Model**)

```
ifdm1 = LinearRegression(fit_intercept = False, n_jobs = None, positive = False)
ifdm2 = Ridge(alpha = 10.0, fit_intercept = False, random_state = None)
ifdm3 = Lasso(alpha = 0.01, fit_intercept = False, random_state = None)
ifdm4 = SVR(C = 10, degree = 3, epsilon = 0.01, gamma = 'auto')
ifdm5 = KNeighborsRegressor(algorithm = 'auto', n_neighbors = 7, p= 1, weights= 'distance')
ifdm6 = DecisionTreeRegressor(max_depth = None, min_samples_leaf = 4,
                             min_samples_split = 4, random_state = None, splitter = 'random')
```


การประเมินโมเดลหลักโดยข้อมูลคุณภาพน้ำของแม่น้ำ 4 สาย (**TEST DATA**)

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	0.9307	1.9189
Ridge Linear Regression	0.9290	1.8957
Lasso Linear Regression	0.9323	1.8846
Support Vector Regression	2.3902	10.5852
K-Neighbors Regressor	1.8474	7.1883
Decision Tree Regressor	1.0807	2.5159

โมเดลและพารามิเตอร์ที่ได้จากวิธีการ **HYPERPARAMETER TUNING**

โมเดลใหม่ (**New Model**)

```
nifdm1 = LinearRegression(fit_intercept = False, n_jobs = None, positive = True)
nifdm2 = Ridge(alpha = 0.01, fit_intercept = False, random_state = None)
nifdm3 = Lasso(alpha = 0.1, fit_intercept = False, random_state = None)
nifdm4 = DecisionTreeRegressor(max_depth = None, min_samples_leaf = 4,
                               min_samples_split = 2, random_state = None, splitter = 'random')
```

การประเมินโมเดลใหม่โดยข้อมูลคุณภาพน้ำของแม่น้ำ 4 สาย (**TEST DATA**)

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	0.9392	1.9433
Ridge Linear Regression	0.9307	1.9187
Lasso Linear Regression	0.9393	1.9001
Decision Tree Regressor	1.0686	2.6133

ข้อแตกต่างระหว่างโมเดลหลักและโมเดลใหม่

โมเดลหลัก (**Main Model**)

```
ifdm1 = LinearRegression(fit_intercept = False, n_jobs = None, positive = False)
ifdm2 = Ridge(alpha = 10.0, fit_intercept = False, random_state = None)
ifdm3 = Lasso(alpha = 0.01, fit_intercept = False, random_state = None)
ifdm4 = SVR(C = 10, degree = 3, epsilon = 0.01, gamma = 'auto')
ifdm5 = KNeighborsRegressor(algorithm = 'auto', n_neighbors = 7, p = 1, weights = 'distance')
ifdm6 = DecisionTreeRegressor(max_depth = None, min_samples_leaf = 4,
                             min_samples_split = 4, random_state = None, splitter = 'random')
```

โมเดลใหม่ (**New Model**)

```
nifdm1 = LinearRegression(fit_intercept = False, n_jobs = None, positive = True)
nifdm2 = Ridge(alpha = 0.01, fit_intercept = False, random_state = None)
nifdm3 = Lasso(alpha = 0.1, fit_intercept = False, random_state = None)
nifdm4 = DecisionTreeRegressor(max_depth = None, min_samples_leaf = 4,
                             min_samples_split = 2, random_state = None, splitter = 'random')
```

การเปรียบเทียบโมเดลหลักที่ใช้ข้อมูลคุณภาพน้ำของแม่น้ำสีสายและข้อมูลคุณภาพน้ำของแม่น้ำน้กตง

Model	I River and Main model		4 Rivers Dataset and Main model	
	Mean Absolute Error	Mean Squared Error	Mean Absolute Error	Mean Squared Error
Linear Regression	0.7987	1.2175	0.9307	1.9189
Ridge Linear Regression	0.7788	1.1566	0.9290	1.8957
Lasso Linear Regression	0.7812	1.1606	0.9323	1.8846
Decision Tree Regressor	0.8264	1.4836	1.0807	2.5159

การเปรียบเทียบโมเดลใหม่และโมเดลหลัก
ที่ใช้ข้อมูลคุณภาพน้ำของแม่น้ำสายและข้อมูลคุณภาพน้ำของแม่น้ำกกดง

Model	4 Rivers Dataset and Main model		4 Rivers Dataset and new model	
	Mean Absolute Error	Mean Squared Error	Mean Absolute Error	Mean Squared Error
Linear Regression	0.9307	1.9189	0.9392	1.9433
Ridge Linear Regression	0.9290	1.8957	0.9307	1.9187
Lasso Linear Regression	0.9323	1.8846	0.9393	1.9001
Decision Tree Regressor	1.0807	2.5159	1.0686	2.6133

COEFFICIENT OF DETERMINATION



```
n_m1.fit(x_train,y_train)
ny1_pred = n_m1.predict(x_test)
print("Mean Absolute Error: ",mean_absolute_error(y_test, ny1_pred))
print("Mean Squared Error: ",mean_squared_error(y_test, ny1_pred, squared = True))
print("Coefficient of determination: ",r2_score(y_test, ny1_pred))
```

Mean Absolute Error: 0.7987065697497872

Mean Squared Error: 1.217500835963755

Coefficient of determination: 0.8079131947702844

การเปรียบเทียบโมเดลหลักที่ใช้ข้อมูลคุณภาพน้ำของแม่น้ำสีสายและข้อมูลคุณภาพน้ำของแม่น้ำนกกดง

Model	I River and Main model	4 Rivers Dataset and Main model
	Coefficient of determination	Coefficient of determination
Linear Regression	0.8079	0.8274
Ridge Linear Regression	0.8175	0.8295
Lasso Linear Regression	0.8168	0.8305
Decision Tree Regressor	0.7662	0.7366

โมเดลที่ควรเลือกใช้สำหรับการคาดการณ์ **CHEMICAL OXYGEN DEMAND**

ในการพิจารณาโมเดลสำหรับการคาดการณ์ Chemical Oxygen Demand มีการประเมินโมเดลด้วย Mean absolute error, Mean squared error และ Coefficient of determination ทำให้ได้ข้อสรุปว่าโมเดลที่เหมาะสมสำหรับการคาดการณ์ควรใช้ชุดข้อมูลของแม่น้ำสีสาย และ มีโมเดลหลักที่สามารถนำไปใช้ในการคาดการณ์ได้ดังต่อไปนี้

1. Lasso Linear Regression
2. Ridge Linear Regression
3. Linear Regression

การหาโมเดลโดย **GRIDSEARCH CV**

1. โมเดลที่หาโดยวิธี **GridsearchCV** มี 4 โมเดลเช่น **Linear Regression, Lasso Linear Regression, Ridge Linear Regression, Decision Tree Regressor**
2. ชุดข้อมูลที่ใช้สำหรับโมเดลเป็นชุดข้อมูลคุณภาพน้ำของแม่น้ำชีสาย
3. การประเมินโมเดลใช้ **Mean Absolute Error** และ **Mean Squared Error**

การหาโมเดลโดย **GRIDSEARCH CV**

```
✓ [239] ifdx_train, ifdx_test, ifdy_train, ifdy_test = train_test_split(x_ifd, y_ifd, test_size = 0.2, random_state = 42)
✓ [240] nparam_grid = {'fit_intercept': [True, False], 'n_jobs': [None, 1, 5, 10], 'positive': [True, False]}
✓ [241] grid_model1 = LinearRegression()
✓ [242] grid_search = GridSearchCV(grid_model1, nparam_grid, cv=5, scoring='neg_mean_absolute_error')
✓ [243] grid_search.fit(ifdx_train, ifdy_train)
```

GridSearchCV

- estimator: LinearRegression
 - LinearRegression

การหาโมเดลโดย **GRIDSEARCH CV**

```
[244] best_params = grid_search.best_params_  
      best_estimator = grid_search.best_estimator_
```

```
[245] best_params
```

```
{'fit_intercept': False, 'n_jobs': None, 'positive': True}
```

```
[246] best_estimator
```

```
LinearRegression  
LinearRegression(fit_intercept=False, positive=True)
```

```
[288] gifdm1 = LinearRegression(fit_intercept = False, positive = True)
```

```
[289] gifdm1.fit(ifdx_train, ifdy_train)  
      gifdm1_pred = gifdm1.predict(ifdx_test)  
      print("Mean Absolute Error: ", mean_absolute_error(ifdy_test, gifdm1_pred))  
      print("Mean Squared Error: ", mean_squared_error(ifdy_test, gifdm1_pred, squared = True))
```

```
Mean Absolute Error: 0.9392536338509349
```

```
Mean Squared Error: 1.9433995609003576
```

โมเดลและพารามิเตอร์ที่ได้จากวิธีการ **GRIDSEARCHCV**

```
gifdm1 = LinearRegression(fit_intercept = False, positive = True)
gifdm2 = Ridge(alpha=0.01, fit_intercept=False)
gifdm3 = Lasso(alpha=0.1, fit_intercept=False)
gifdm4 = DecisionTreeRegressor(min_samples_leaf=4, min_samples_split=3, splitter='random')
```

การประเมินโมเดลของ **GRIDSEARCHCV** โดยข้อมูลคุณภาพน้ำของแม่น้ำ 4 สาย

Model	Mean Absolute Error	Mean Squared Error
Linear Regression	0.9392	1.9433
Ridge Linear Regression	0.9307	1.9187
Lasso Linear Regression	0.9393	1.9001
Decision Tree Regressor	1.0437	2.3497

ข้อแตกต่างระหว่างโมเดลหลักและโมเดลของ **GRIDSEARCHCV**

โมเดลหลัก (**Main Model**)

```
ifdm1 = LinearRegression(fit_intercept = False, n_jobs = None, positive = False)
ifdm2 = Ridge(alpha = 10.0, fit_intercept = False, random_state = None)
ifdm3 = Lasso(alpha = 0.01, fit_intercept = False, random_state = None)
ifdm4 = SVR(C = 10, degree = 3, epsilon = 0.01, gamma = 'auto')
ifdm5 = KNeighborsRegressor(algorithm = 'auto', n_neighbors = 7, p = 1, weights = 'distance')
ifdm6 = DecisionTreeRegressor(max_depth = None, min_samples_leaf = 4,
                             min_samples_split = 4, random_state = None, splitter = 'random')
```

โมเดลของ **GridsearchCV (GridsearchCV Model)**

```
#LinearRegression(fit_intercept = False, positive = True)
#Ridge(alpha=0.01, fit_intercept=False)
#gifdm3 = Lasso(alpha=0.1, fit_intercept=False)
#gifdm4 = DecisionTreeRegressor(min_samples_leaf=4, splitter='random')
```

การเปรียบเทียบโมเดลหลักและโมเดลของ **GRIDSEARCHCV**
ที่ใช้ข้อมูลคุณภาพน้ำของแม่น้ำสีสาย

Model	Main Model (Hyperparameter tuning)		GridsearchCV Model	
	Mean Absolute Error	Mean Squared Error	Mean Absolute Error	Mean Squared Error
Linear Regression	0.9307	1.9189	0.9392	1.9433
Ridge Linear Regression	0.9290	1.8957	0.9307	1.9187
Lasso Linear Regression	0.9323	1.8846	0.9393	1.9001
Decision Tree Regressor	1.0807	2.5159	1.0922	2.6041

ข้อสรุปผล

1. ในการพิจารณาโมเดลของ Hyperparameter Tuning สำหรับการคาดการณ์ Chemical Oxygen Demand มีการประเมินโมเดลด้วย Mean absolute error, Mean squared error และ Coefficient of determination ทำให้ได้ข้อสรุปว่าโมเดลที่เหมาะสมสำหรับการคาดการณ์ควรใช้ชุดข้อมูลของแม่น้ำสีสาย และ มีโมเดลหลักที่สามารถนำไปใช้ในการคาดการณ์ **เช่น Lasso Linear Regression Ridge Linear Regression และ Linear Regression**

ข้อสรุปผล

2. Mean squared error ของโมเดลหลักจากวิธีการ Hyperparameter Tuning มีค่าน้อยกว่า Mean squared error ของโมเดล GridsearchCV ดังนั้นการคาดการณ์ค่า COD ควรใช้โมเดลหลักจากวิธีการ Hyperparameter Tuning เช่น Lasso Linear Regression Ridge Linear Regression และ Linear Regression เพื่อให้ค่า COD ที่คาดการณ์แม่นยำมากที่สุด การเลือกโมเดลไปใช้ขึ้นอยู่กับค่าความคลาดเคลื่อนในการคาดการณ์ที่ยอมรับได้ของหน่วยงานภาครัฐและเอกชน

การนำไปใช้

1. โมเดลที่ใช้สำหรับการคาดการณ์ค่า COD สามารถนำไปใช้โดยมีตัวแปรต้น 17 ตัวแปรและตัวแปรตาม 1 ตัวแปรเป็นตัวแปรเกี่ยวกับคุณภาพน้ำที่เป็นประโยชน์ต่อภาครัฐในการศึกษาและตรวจสอบคุณภาพน้ำของแม่น้ำเพราะในประเทศไทยมีแม่น้ำหลายสายซึ่งเป็นส่วนสำคัญในด้านเศรษฐกิจเช่น การเกษตรกรรม การประมง และ การอุปโภคบริโภคในครัวเรือน

การนำไปใช้

2. โมเดลที่มี 18 ตัวแปรทำให้ภาครัฐสามารถเก็บข้อมูลในตัวแปรดังกล่าวได้เพื่อลดเวลาในการเก็บข้อมูลตามจำนวนตัวแปรที่มากขึ้น และ ภาครัฐสามารถนำข้อมูลที่มีรัฐมีไปใช้ในการประเมินคุณภาพน้ำและคาดการณ์ค่า COD ของแม่น้ำแต่ละสายได้

การนำไปใช้

3. โมเดลเป็นประโยชน์ต่อภาคเอกชนเช่น โรงงานอุตสาหกรรม โรงไฟฟ้า โรงแรม และ องค์การทางธุรกิจอื่นๆที่ต้องการนำน้ำไปใช้ในองค์กรเพื่อการบริการ การผลิต การบำบัดน้ำเสียเพื่อนำน้ำกลับแหล่งน้ำธรรมชาติเพื่อให้การใช้น้ำของภาคเอกชนเป็นไปตามมาตรฐานที่กำหนดเพื่อเป็นการรักษาสีสิ่งแวดล้อมเช่น การปล่อยน้ำของโรงงานกลับสู่แหล่งน้ำธรรมชาติ

ข้อเสนอแนะ

1. การเก็บข้อมูลของแม่น้ำสายอื่นเพื่อให้โมเดลสามารถคาดการณ์ค่า COD แม่นยำขึ้น
2. การเก็บข้อมูลจากชุดข้อมูลของฤดูร้อนเป็นชุดข้อมูลใน 1 ปีเพราะแต่ละประเทศมีฤดูกาลที่แตกต่างกันเช่น ประเทศไทยมี 3 ฤดูกาล ประเทศเกาหลีใต้มี 4 ฤดูกาลทำให้โมเดลสามารถคาดการณ์ค่า COD เมื่อนำไปใช้กับข้อมูลของแม่น้ำสายอื่นและไม่มีข้อจำกัดการใช้โมเดลสำหรับในฤดูกาลที่มีการเก็บข้อมูล

ข้อเสนอแนะ

3. การปรับปรุงโมเดลให้สามารถคาดการณ์ค่า COD ได้ในอนาคตที่อ้างอิงจากชุดข้อมูลที่มีอยู่ในปัจจุบัน
4. พิจารณาเรื่องการเก็บข้อมูล 18 ตัวแปรว่าการเก็บข้อมูลให้น้อยกว่า 18 ตัวแปรสามารถทำได้หรือไม่เพื่อลดเวลาในการเก็บข้อมูลเพราะแหล่งเก็บข้อมูลมีการเผยแพร่ชุดข้อมูลคุณภาพน้ำที่มีตัวแปรแตกต่างกันโดยต้องคำนึงถึงปัญหา Overfitting และ Underfitting เมื่อนำโมเดลไปใช้



END OF PRESENTATION