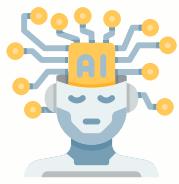


Machine Learning

Water Prediction



Programs used to make projects

Orange data mining

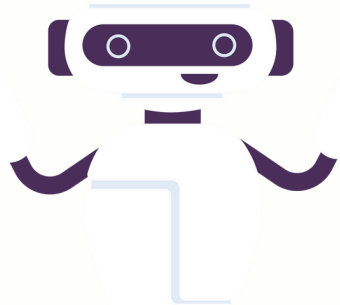
- เป็นเครื่องมือวิเคราะห์ข้อมูล และ machine learning ที่มี ส่วนต่อประสานแบบกราฟิก (GUI) ทำให้ผู้ใช้สามารถสร้าง ขั้นตอนการทำงาน (workflow) เพื่อวิเคราะห์ข้อมูลได้อย่างสะดวก
- จุดเด่นคือ ใช้งานง่าย เหมาะสำหรับผู้เริ่มต้น จนถึงผู้เชี่ยวชาญที่ต้องการ สืบรวจและวิเคราะห์ข้อมูลเชิงลึก โดยไม่ต้องเขียนโค้ดที่ซับซ้อน

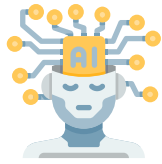
Excel

- โปรแกรมสเปรดชีต ที่ใช้กันอย่างแพร่หลายในการ จัดการข้อมูลเชิงตาราง คำนวณ และวิเคราะห์ข้อมูล
- มีฟังก์ชันหลากหลายสำหรับการสร้างสูตร คำนวณทางสถิติ สร้างกราฟ และ สรุปผลข้อมูล เหมาะสำหรับงานที่ต้องการ จัดการข้อมูลในปริมาณมาก และ นำเสนอข้อมูลอย่างมีประสิทธิภาพ

R Studio

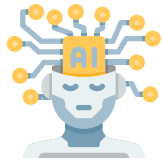
- สภาพแวดล้อมการพัฒนา (IDE) สำหรับภาษา R ซึ่งเป็นภาษาโปรแกรมที่เน้นด้าน สถิติ และ การวิเคราะห์ข้อมูล
- มีเครื่องมือที่ช่วยในการ เขียนโค้ด การแก้ไขจุดบกพร่อง (debug) และการแสดงผลลัพธ์ ทำให้การทำงานกับ R มีประสิทธิภาพสูง
- เหมาะสำหรับ นักสถิติ นักวิทยาศาสตร์ข้อมูล และผู้ที่ต้องการ วิเคราะห์ข้อมูลเชิงลึก ด้วยภาษา R



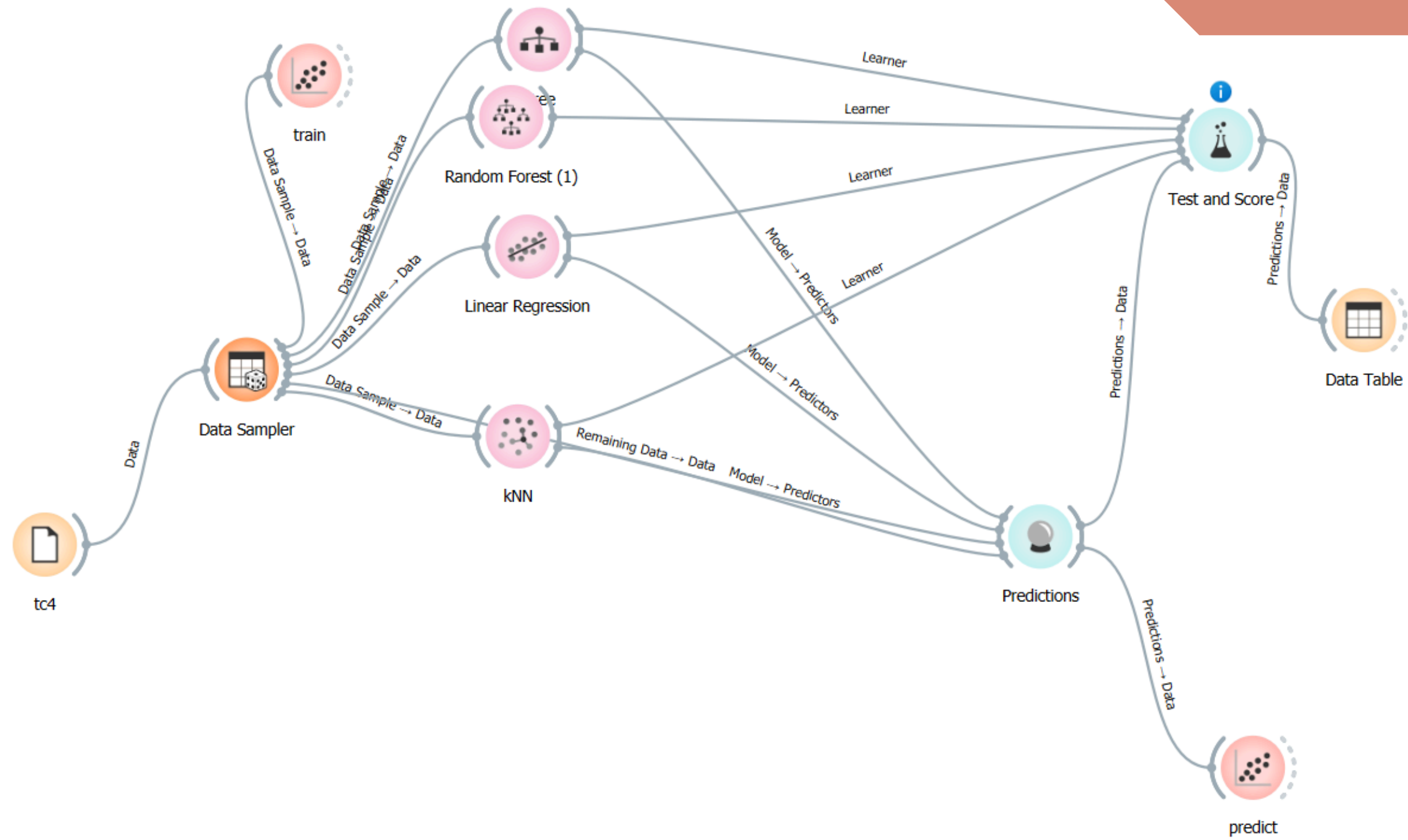


Orange Data Mining





EXAMPLE

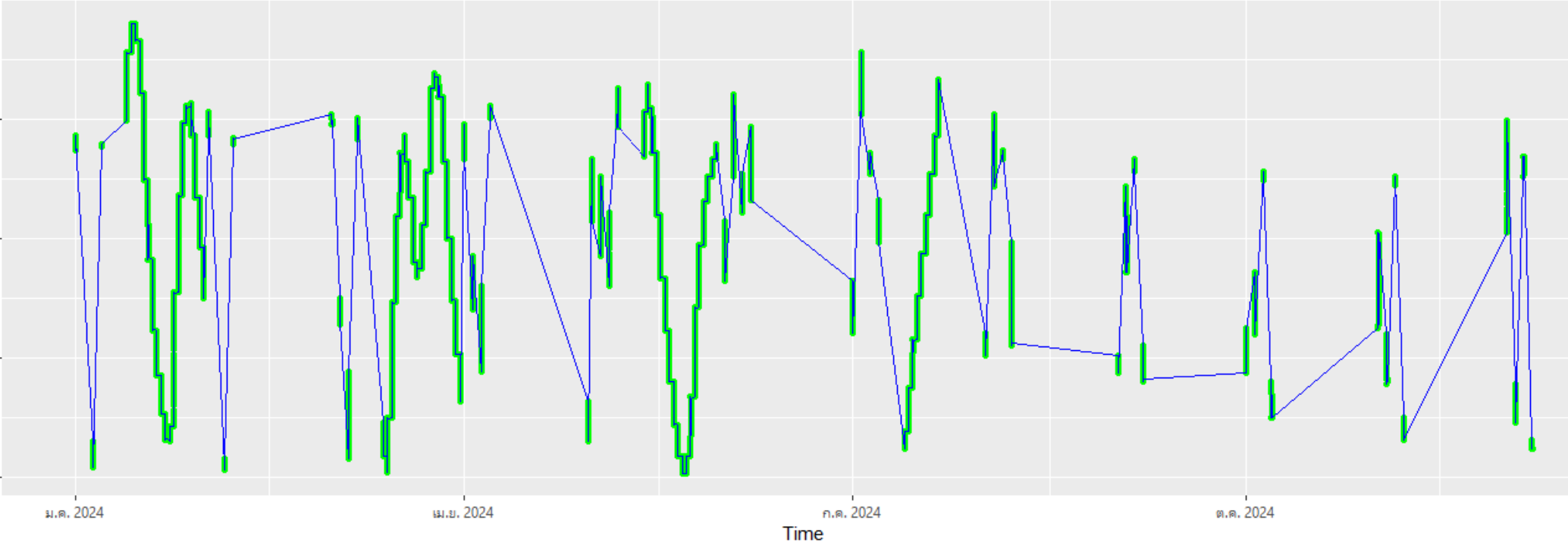


A	B	C
station_id	datetime	value
6	48:08.0	0.69
6	48:08.0	0.69
6	49:08.0	0.7
6	49:08.0	0.7
6	50:08.0	
6	50:08.0	
6	51:08.0	
6	51:08.0	
6	52:08.0	
6	52:08.0	
6	53:08.0	
6	53:08.0	
6	54:08.0	
6	54:08.0	
6	55:08.0	
6	55:08.0	

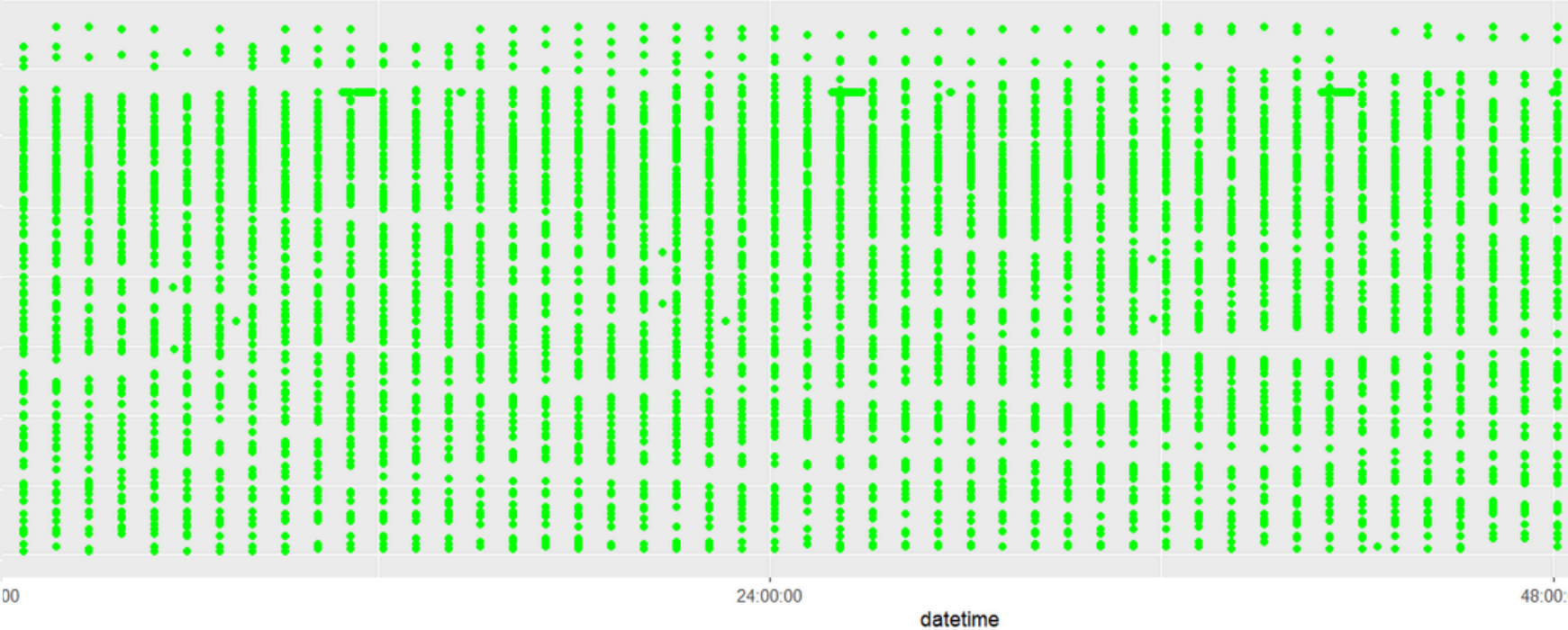
- Clean data Datetime Format to
- YYYY-MM-DD , HH:MM:S

A	B	C
station_id	value	datetime
6	0.69	1/1/2024 0:48
6	0.69	1/1/2024 0:48
6	0.7	1/1/2024 0:49
6	0.7	1/1/2024 0:49
6	0.7	1/1/2024 0:50
6	0.7	1/1/2024 0:50
6	0.71	1/1/2024 0:51
6	0.71	1/1/2024 0:51

Scater Plot with Time Series Line



Normalized Data



STEP 2

- # Do Z-score with colum -> valu

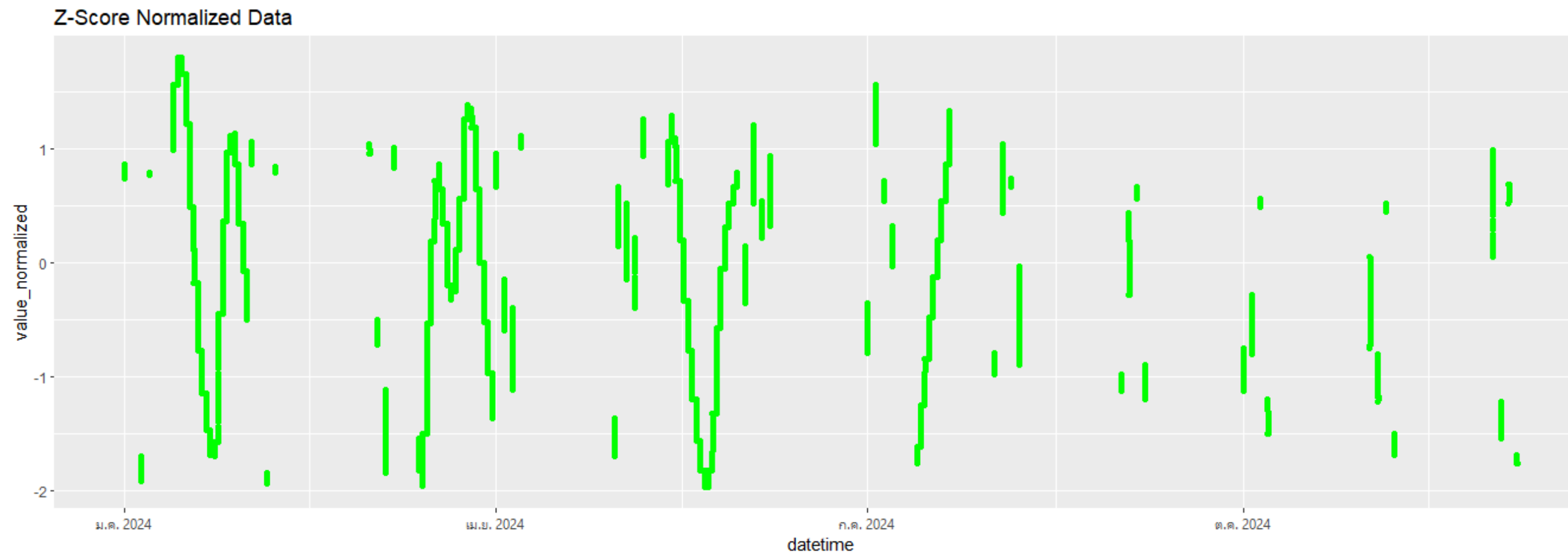
```
# Do Z-score with colum -> value -----
mydata_normalized <- mydata %>%
  mutate(value_normalized = z_score_normalization(value))

#add columns for low normal high

mydata_normalized <- mydata_normalized %>%
  mutate(water_level_category = case_when(
    value_normalized < -1 ~ "Low",
    between(value_normalized, -1, 1) ~ "Normal",
    value_normalized > 1 ~ "High",
    TRUE ~ "Undefined"
  ))

# output
print(mydata_normalized)

#plot
```



```

# Do Z-score with colum -> value -----
mydata_normalized <- mydata %>%
  mutate(value_normalized = z_score_normalization(value))

#add columns for low normal high

mydata_normalized <- mydata_normalized %>%
  mutate(water_level_category = case_when(
    value_normalized < -1 ~ "Low",
    between(value_normalized, -1, 1) ~ "Normal",
    value_normalized > 1 ~ "High",
    TRUE ~ "Undefined"
  ))

# output
print(mydata_normalized)

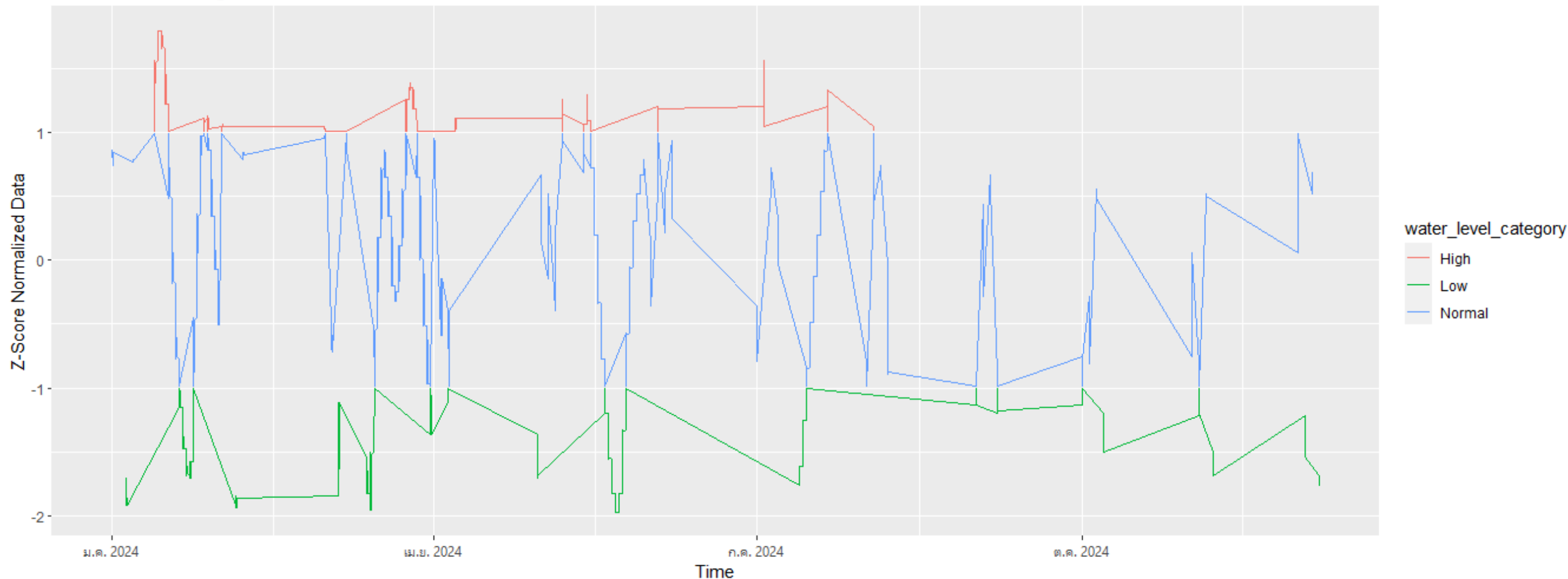
#plot

```

STEP 2

- # Do Z-score with colum -> val
 - # Group by columns Value
- > HIGH NORMAL LOW

Water Level Categories



STEP 3

File : TC4_tranform.csv and set
value_normalized to target

The screenshot shows the 'tc4 - Orange' window. The 'Source' section has 'File: TC4\tc4_tranform.csv' selected. The 'File Type' is set to 'Automatically detect type'. The 'Info' section displays: 13841 instances, 8 features (no missing values), Data has no target variable, and 0 meta attributes. The 'Columns (Double click to edit)' table is as follows:

	Name	Type	Role	Values
1	station_id	N numeric	feature	
2	value	N numeric	skip	
3	datetime	T datetime	feature	
4	hour	N numeric	skip	0
5	minute	N numeric	feature	
6	second	N numeric	feature	
7	value_normali...	N numeric	target	
8	water_level_cat...	C categorical	feature	High, Low, Normal

At the bottom, there are 'Reset' and 'Apply' buttons, and a 'Browse documentation datasets' link. The status bar at the very bottom shows icons for menu, help, and a file icon with '13.8k'.

STEP 3

SPLIT DATA 80:20

Tran : 11073 instance

6 variables

Test : 2768 instance

6 variables

Data Sample: tc4 tranfrom: 11073 instances, 6 variables

Features: 5 (1 categorical, 3 numeric, 1 time) (no missing values)

Target: numeric

	value_normalized	station_id	datetime	minute	second	water_level_category
1	-0.178169	6	2024-07-17 ...	51	8	Normal
2	0.611823	6	2024-06-07 ...	20	8	Normal
3	1.25818	6	2024-07-03 ...	41	8	High
4	-1.61452	6	2024-03-13 ...	10	8	Low
5	1.31204	6	2024-01-16 ...	53	8	High

Remaining Data: tc4 tranfrom: 2768 instances, 6 variables


Features: 5 (1 categorical, 3 numeric, 1 time) (no missing values)

Target: numeric

	value_normalized	station_id	datetime	minute	second	water_level_category
1	-0.824525	6	2024-11-03 ...	2	8	Normal
2	-1.90179	6	2024-05-22 ...	23	8	Low
3	0.378416	6	2024-05-01 ...	32	8	Normal
4	1.1325	6	2024-05-07 ...	10	8	High
5	1.33	6	2024-07-21 ...	56	45	High



Example datatable and info

 Data Info...

?

×

Data table properties

Name: tc4_tranfrom

Size: 11073 rows, 6 columns


Features: 1 categorical, 4 numeric

Targets: numeric target variable


Additional attributes

≡

?



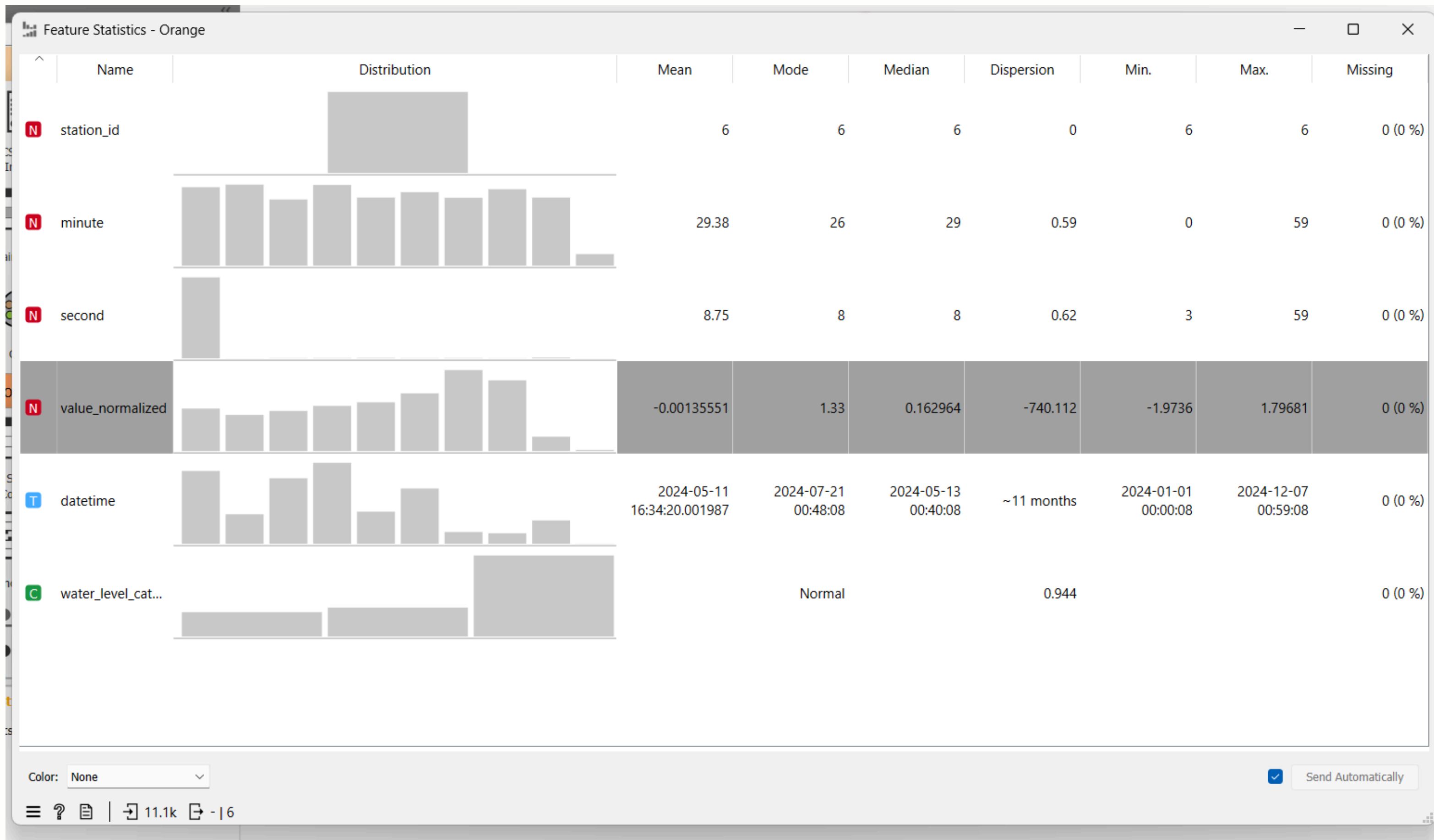
|

 11.1k

value_normalized	station_id	datetime	minute	second	water_level_category
-0.178169	6	2024-07-17 00:...	51	8	Normal
0.611823	6	2024-06-07 00:...	20	8	Normal
1.25818	6	2024-07-03 00:...	41	8	High
-1.61452	6	2024-03-13 00:...	10	8	Low
1.31204	6	2024-01-16 00:...	53	8	High
1.47363	6	2024-07-03 00:...	48	8	High
0.755458	6	2024-01-17 00:...	36	8	Normal
-1.9377	6	2024-03-14 00:...	10	8	Low
1.20432	6	2024-07-03 00:...	46	8	High
0.522051	6	2024-06-05 00:...	56	8	Normal
0.180919	6	2024-01-30 00:...	18	8	Normal
-1.88383	6	2024-01-05 00:...	48	8	Low
1.04273	6	2024-01-13 00:...	4	8	High
0.827275	6	2024-03-18 00:...	18	8	Normal
-1.5427	6	2024-11-07 00:...	14	8	Low
1.33	6	2024-07-21 00:...	26	19	High
0.342508	6	2024-07-19 00:...	23	8	Normal
-0.96816	6	2024-05-19 00:...	25	8	Normal
-1.72224	6	2024-07-13 00:...	11	8	Low
0.0552381	6	2024-03-28 00:...	53	8	Normal
0.988865	6	2024-01-27 00:...	5	8	Normal
0.952956	6	2024-05-13 00:...	25	8	Normal
1.36591	6	2024-07-03 00:...	32	8	High
-0.375666	6	2024-10-03 00:...	11	8	Normal
1.18636	6	2024-05-07 00:...	13	8	High
1.07864	6	2024-06-03 00:...	47	8	High



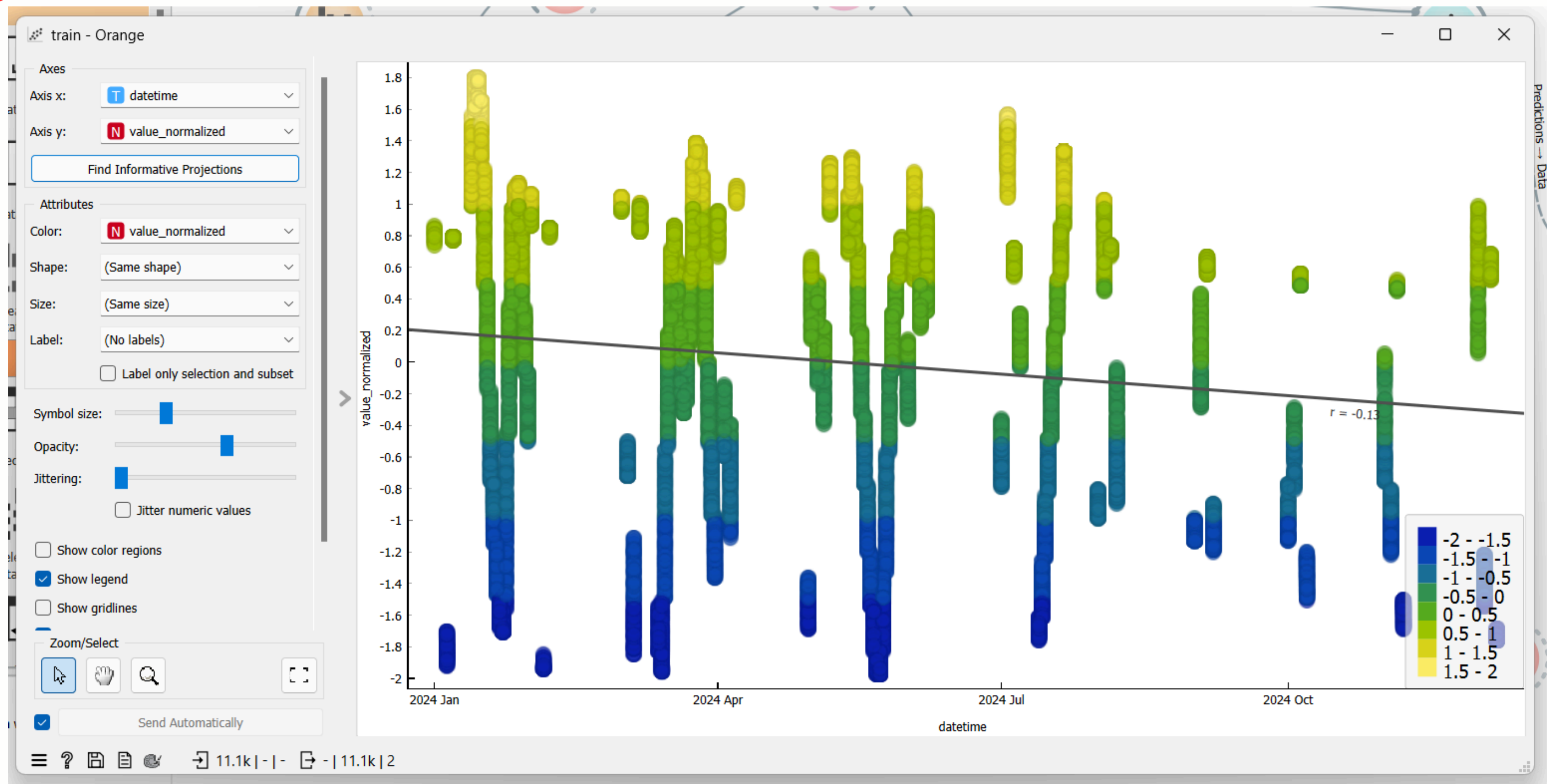
Basic statistics





Plot for scatter

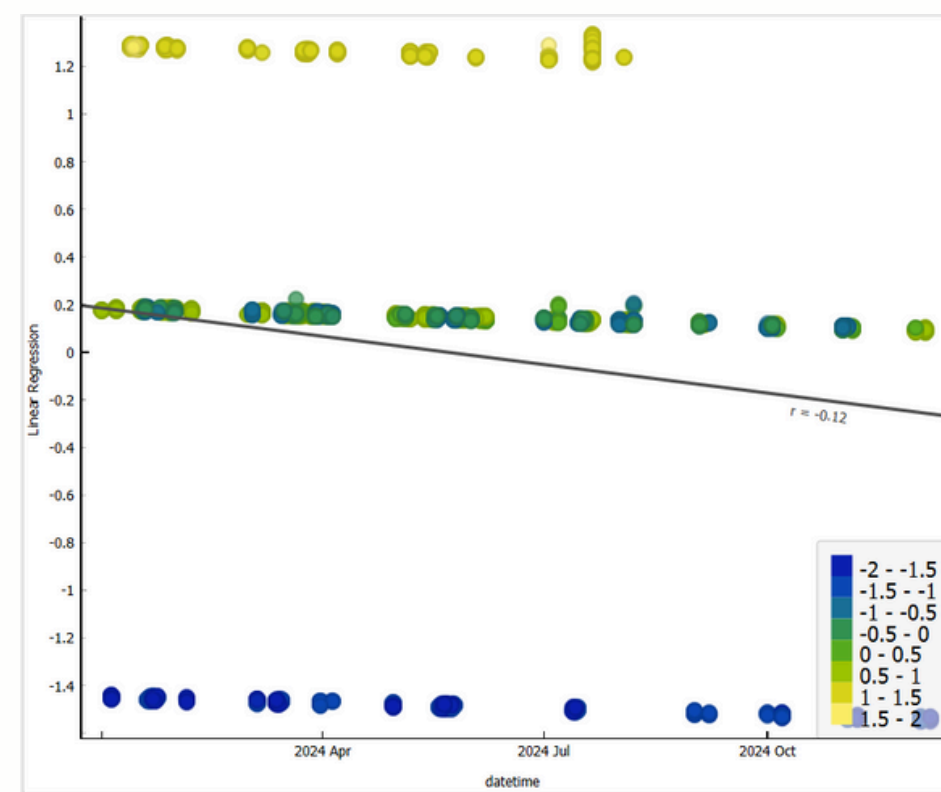
11



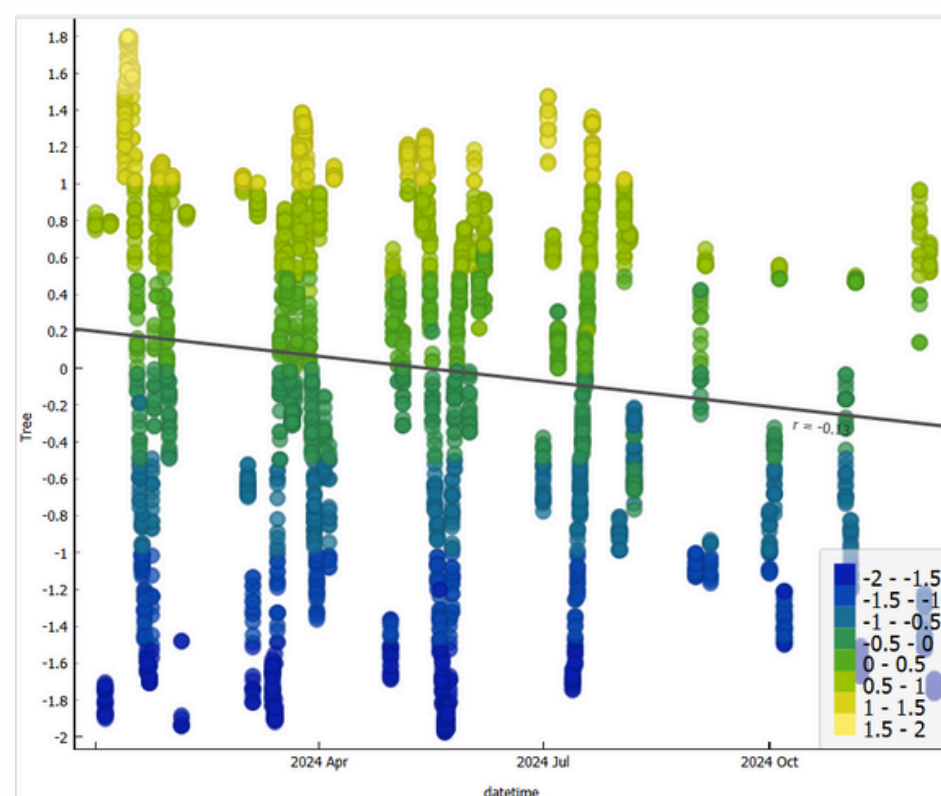


Plots for Scatter all model

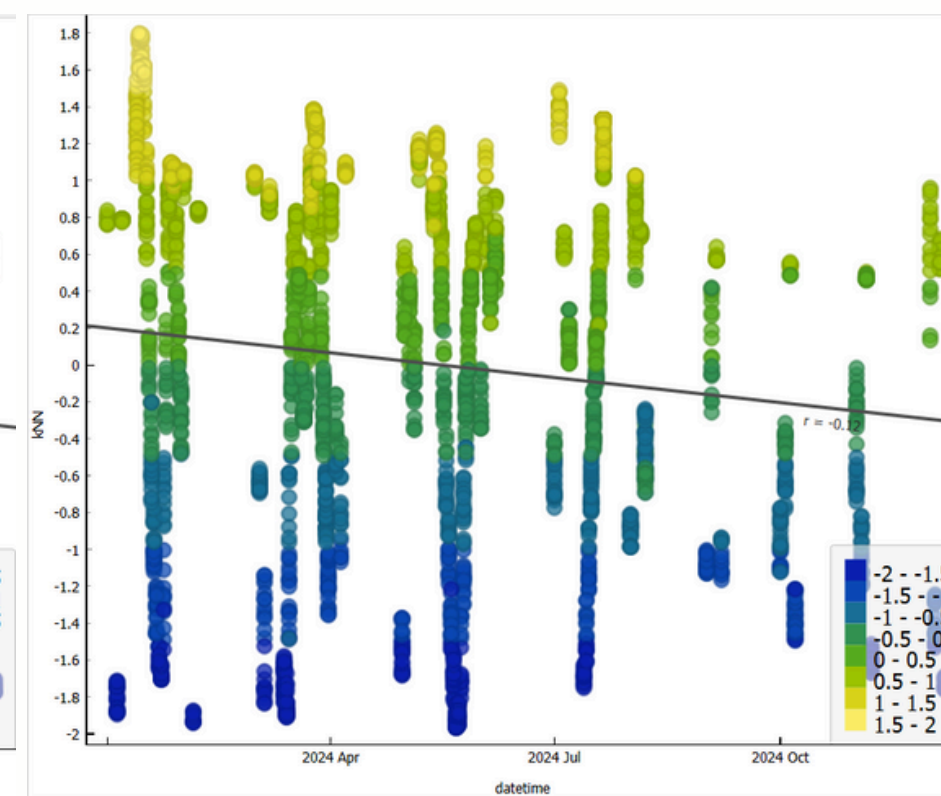
Linear



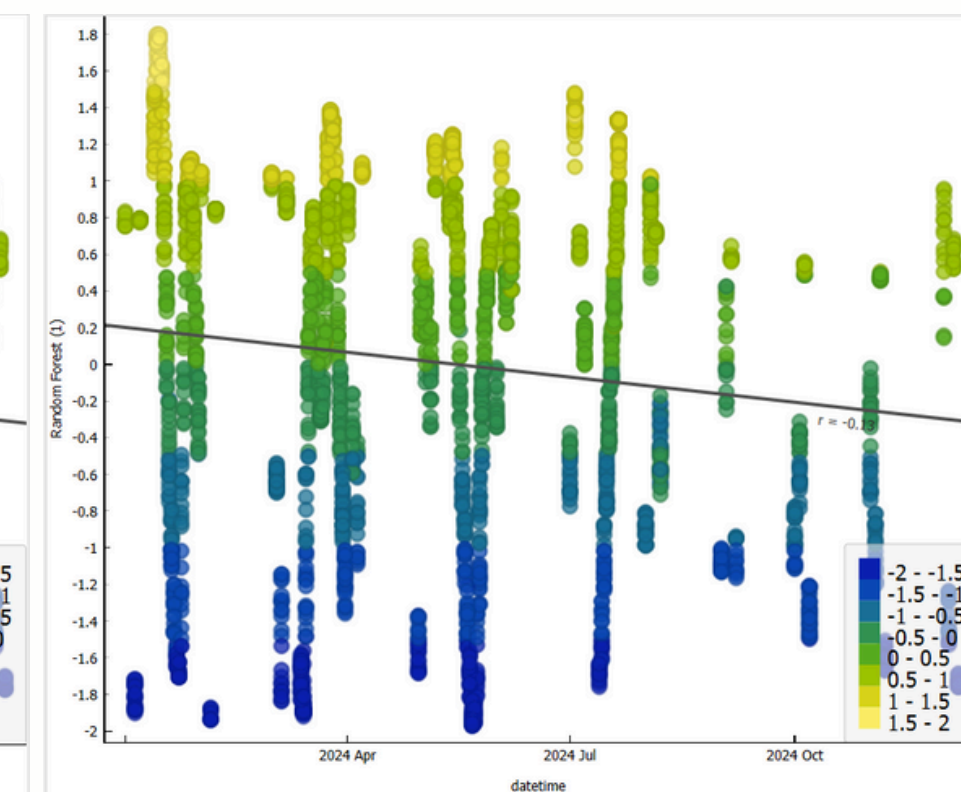
tree



Knn



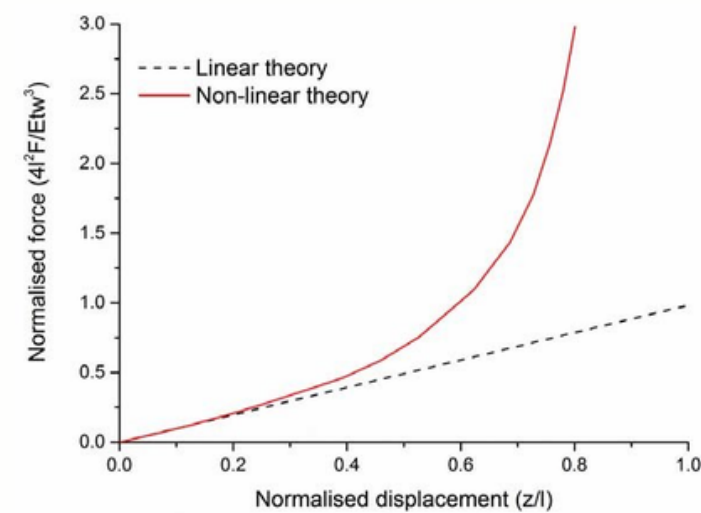
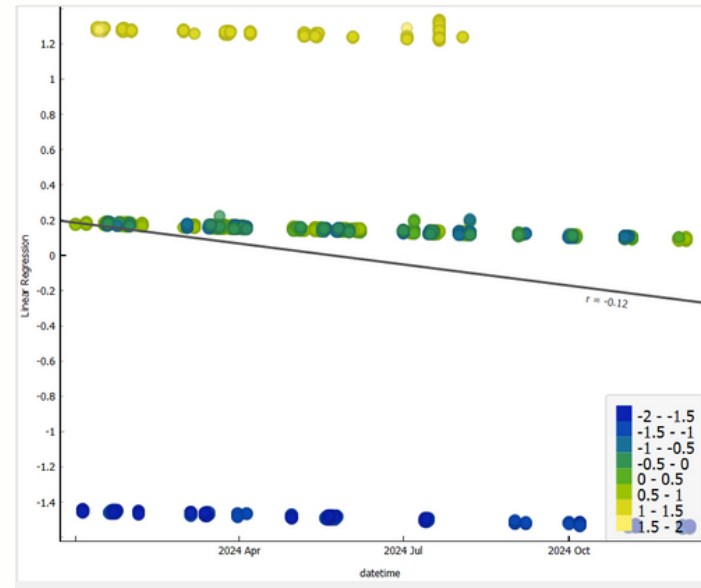
random forest





Linear Theory

Linear



ทฤษฎีเชิงเส้น (Linear Theory)

ทฤษฎีเชิงเส้น เป็นแนวความคิดทางคณิตศาสตร์ที่ใช้ศึกษาความสัมพันธ์ระหว่างตัวแปรต่างๆ โดยอาศัยสมการเชิงเส้น

ประเภทของทฤษฎีเชิงเส้น

สมการเชิงเส้น: เป็นสมการที่มีรูปแบบ $y = mx + b$ โดยที่:

y คือ ตัวแปรตาม

x คือ ตัวแปรอิสระ

m คือ ความชัน

b คือ ระยะตัดแกน y

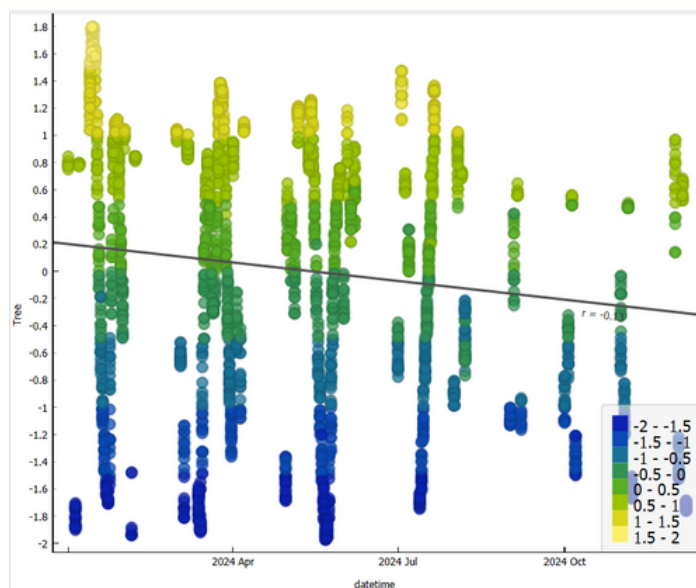
ตัวอย่างการประยุกต์ใช้ทฤษฎีเชิงเส้น

- การวิเคราะห์ความสัมพันธ์: ใช้หาความสัมพันธ์ระหว่างตัวแปรต่างๆ เช่น ความสัมพันธ์ระหว่างยอดขายกับค่าโฆษณา
- การสร้างแบบจำลอง: ใช้สร้างแบบจำลองทางคณิตศาสตร์เพื่อจำลองระบบต่างๆ เช่น แบบจำลองการเติบโตของประชากร
- การแก้ปัญหทางเรขาคณิต: ใช้แก้ปัญหทางเรขาคณิต เช่น การหาจุดตัดของเส้นตรง



Tree Theory

Tree



ทฤษฎีต้นไม้ (Tree Theory) สำหรับการสร้างแบรนด์ เปรียบเสมือนแนวคิดในการสร้างรากฐานที่มั่นคงให้กับแบรนด์ เปรียบเสมือนต้นไม้ที่แข็งแรง

หลักการทำงาน

การสร้างต้นไม้: อัลกอริทึมจะเริ่มต้นด้วยชุดข้อมูลทั้งหมดและค้นหาตัวแปรที่จะแบ่งข้อมูลออกเป็นสองกลุ่มย่อยที่มีความบริสุทธิ์ของคลาสสูงสุด

การแบ่งข้อมูล: ข้อมูลจะถูกแบ่งออกเป็นสองกลุ่มย่อยตามค่าของตัวแปรที่เลือก

การวนซ้ำ: ขั้นตอนการแบ่งข้อมูลจะถูกทำซ้ำกับกลุ่มย่อยแต่ละกลุ่มจนกว่าจะได้กลุ่มย่อยที่มีความบริสุทธิ์ของคลาสสูงสุด หรือจนกว่าจะถึงเกณฑ์การหยุด

ผลลัพธ์: โมเดล Tree จะแสดงผลในรูปแบบของต้นไม้ที่มีกิ่งก้านสาขา แต่ละกิ่งก้านแสดงถึงตัวแปรที่ใช้ในการตัดสินใจ แต่ละใบแสดงถึงกลุ่มย่อยของข้อมูล

ประเภทของ Tree

Decision Tree: ใช้สำหรับงานการจำแนกประเภท

Regression Tree: ใช้สำหรับงานการถดถอย

ตัวอย่างการใช้งาน

การวิเคราะห์ข้อมูลลูกค้า: คาดการณ์ว่าลูกค้าแต่ละรายจะซื้อสินค้าหรือไม่

การวิเคราะห์ความเสี่ยง: คาดการณ์ว่าผู้กู้แต่ละรายจะผิดนัดชำระหนี้หรือไม่

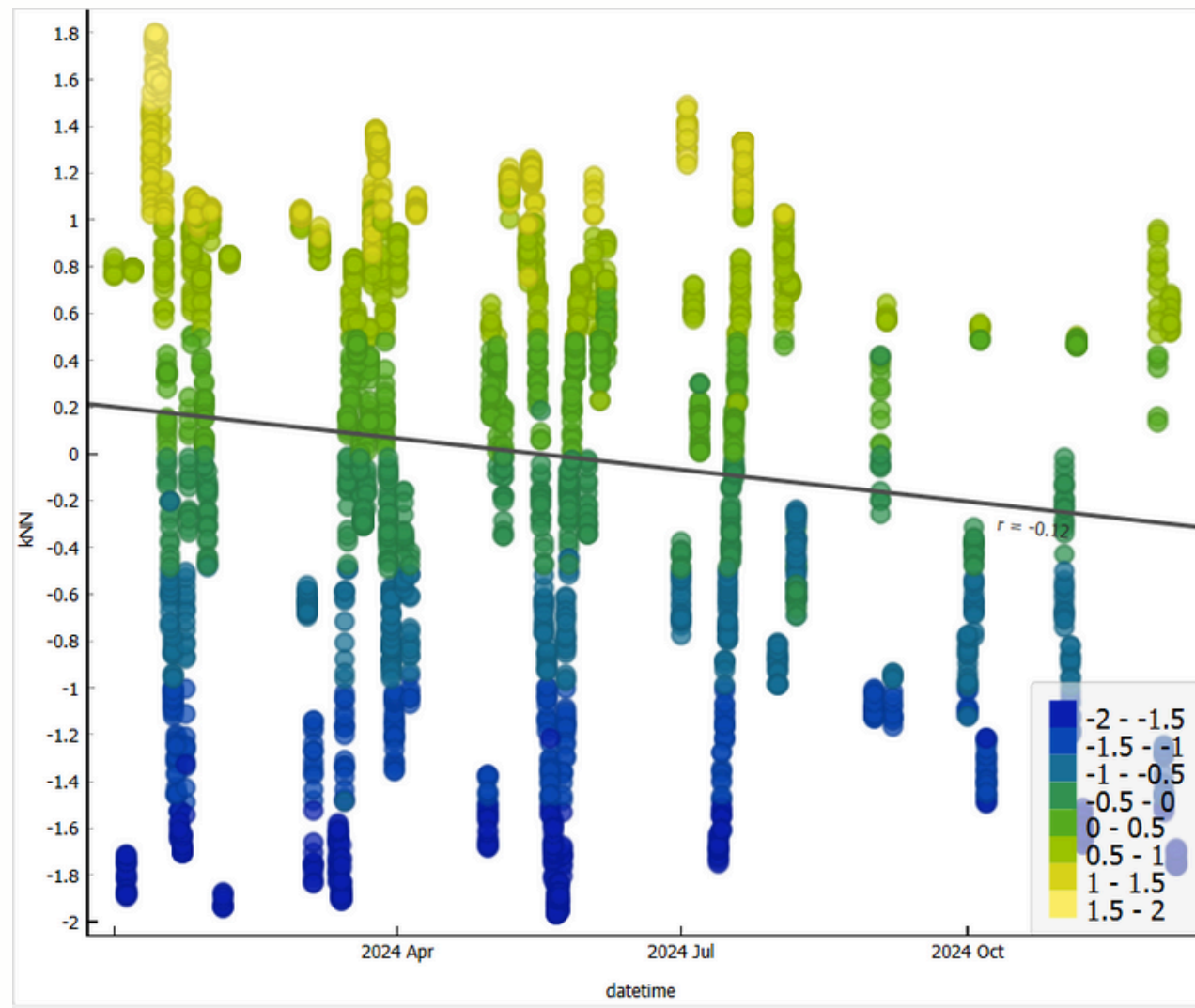
การคาดการณ์ราคาอสังหาริมทรัพย์: คาดการณ์ราคาขายของอสังหาริมทรัพย์แต่ละแห่ง



Knn Theory

5

Knn

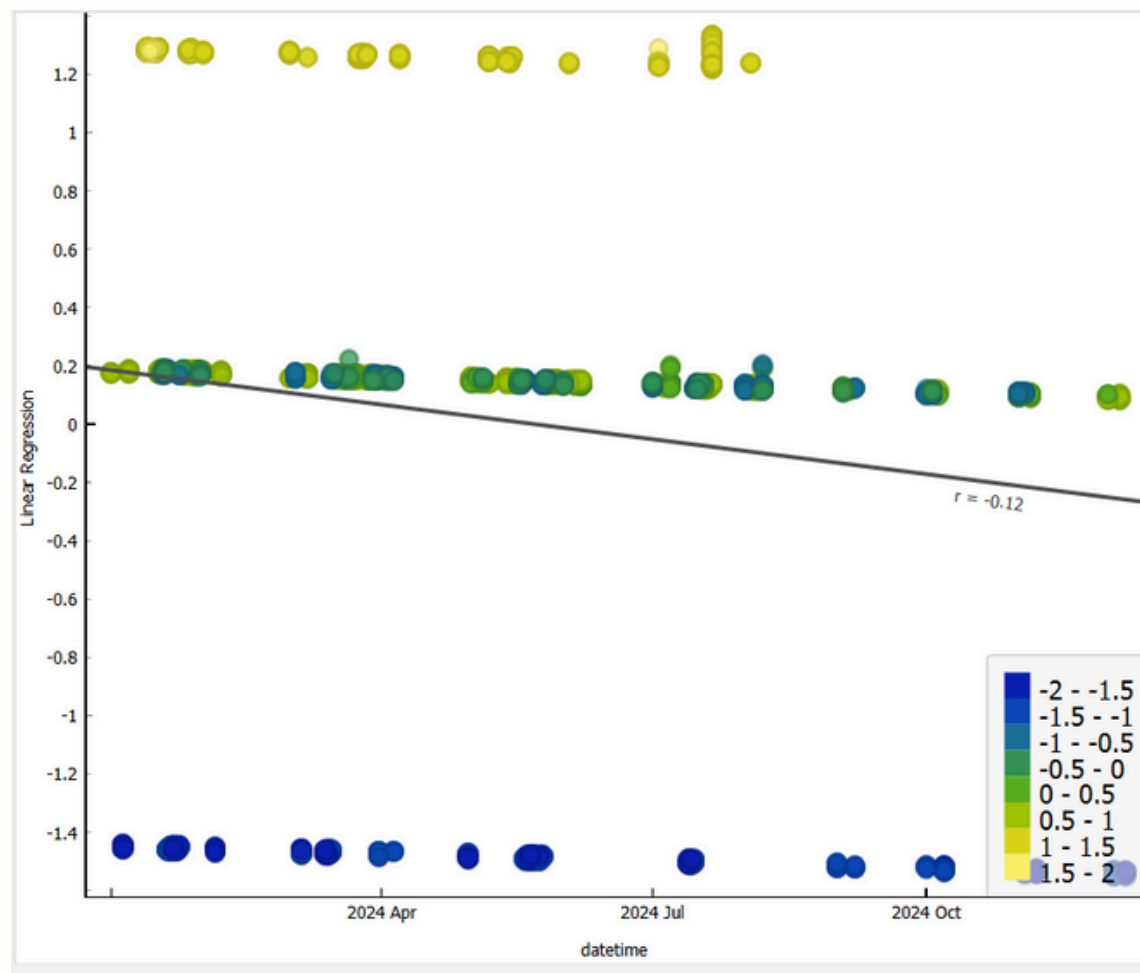


K-Nearest Neighbors (Knn Theory) เป็นอัลกอริทึมการเรียนรู้ของเครื่องแบบไม่ใช้โมเดล (Non-parametric) ที่ใช้สำหรับงานจำแนกประเภท (Classification) และการถดถอย (Regression) โดยใช้วิธีการเปรียบเทียบข้อมูลใหม่กับข้อมูลที่มีอยู่แล้ว



Random forest Theory

random forest



ทฤษฎี Random Forest (Random Forest Theory) คือ เทคนิคการเรียนรู้ของเครื่องจักรประเภท Ensemble Learning ที่นำเอา Decision Tree (ต้นไม้ตัดสินใจ) หลาย ๆ ต้นมาทำงานร่วมกัน



Standard score (z-scores)

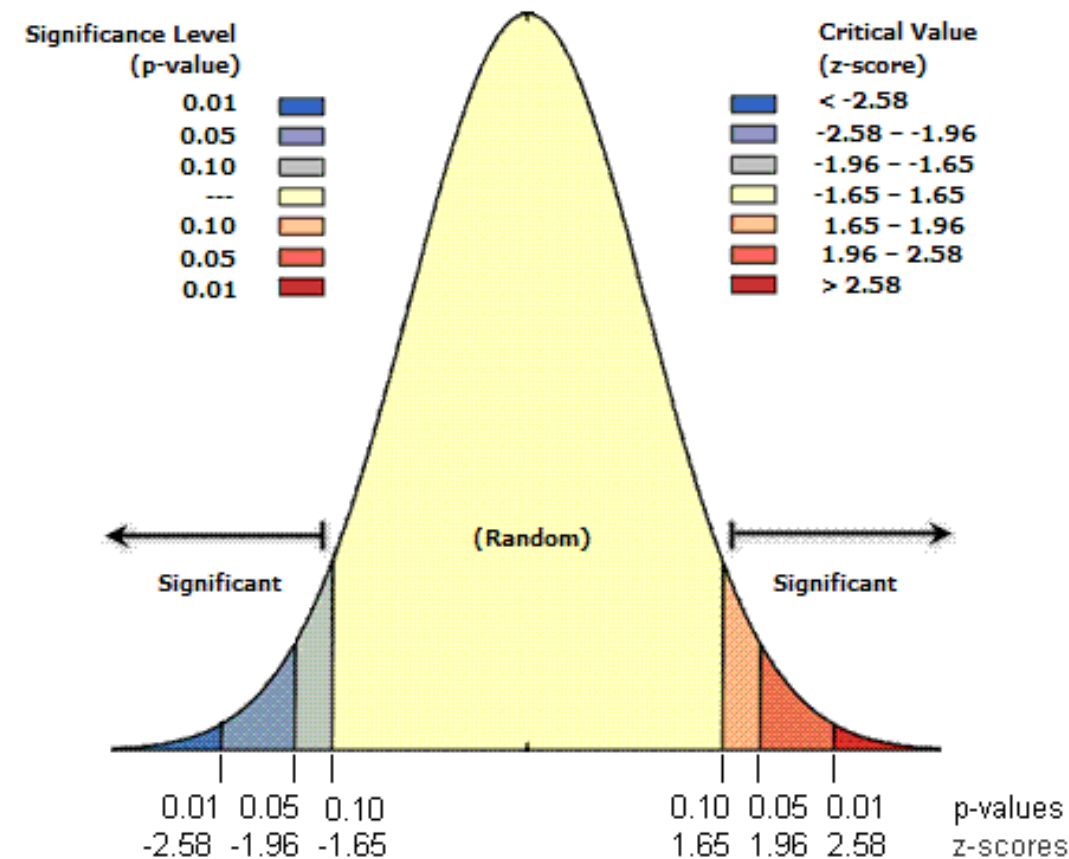
Variance

$$\sigma^2 = \frac{\sum (X - \bar{X})^2}{N}$$

Standard deviation

$$\sigma = \sqrt{\frac{\sum (X - \bar{X})^2}{N}}$$

σ = ค่าเบี่ยงเบนมาตรฐาน
 σ^2 = ค่าแปรปรวน
 X = ค่าตัวแปร
 \bar{X} = ค่าเฉลี่ย
 N = จำนวนตัวแปรทั้งหมด



- ช่วยให้เปรียบเทียบคะแนนจากชุดข้อมูลที่ต่างกันได้
- คะแนน Z มีค่าเฉลี่ย = 0 และค่าเบี่ยงเบนมาตรฐาน = 1
- ช่วยให้เข้าใจว่าคะแนนดิบอยู่ห่างจากค่าเฉลี่ยกี่หน่วยค่าเบี่ยงเบนมาตรฐาน



Predict Water Levels

tc4 - Orange

Source

☒ File: TC4\tc4_tranfrom.csv

☐ URL:

File Type

Automatically detect type

Info

13841 instances
8 features (no missing values)
Data has no target variable.
0 meta attributes

Columns (Double click to edit)

	Name	Type	Role	Values
1	station_id	N numeric	feature	
2	value	N numeric	skip	
3	datetime	T datetime	feature	
4	hour	N numeric	skip	0
5	minute	N numeric	feature	
6	second	N numeric	feature	
7	value_normalized	N numeric	feature	
8	water_level_ca...	C categorical	target	High, Low, Normal

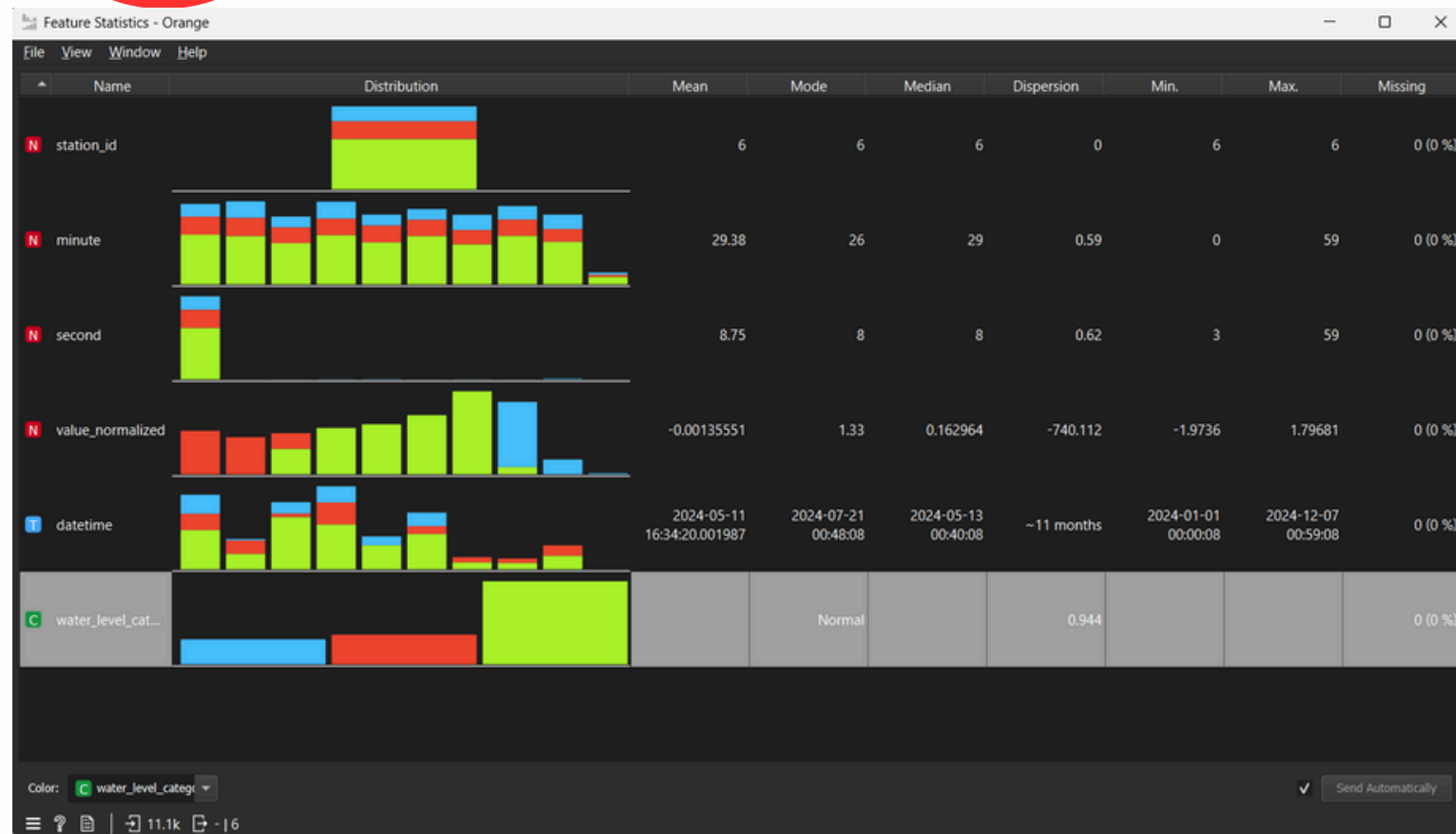
Reset

13.8k

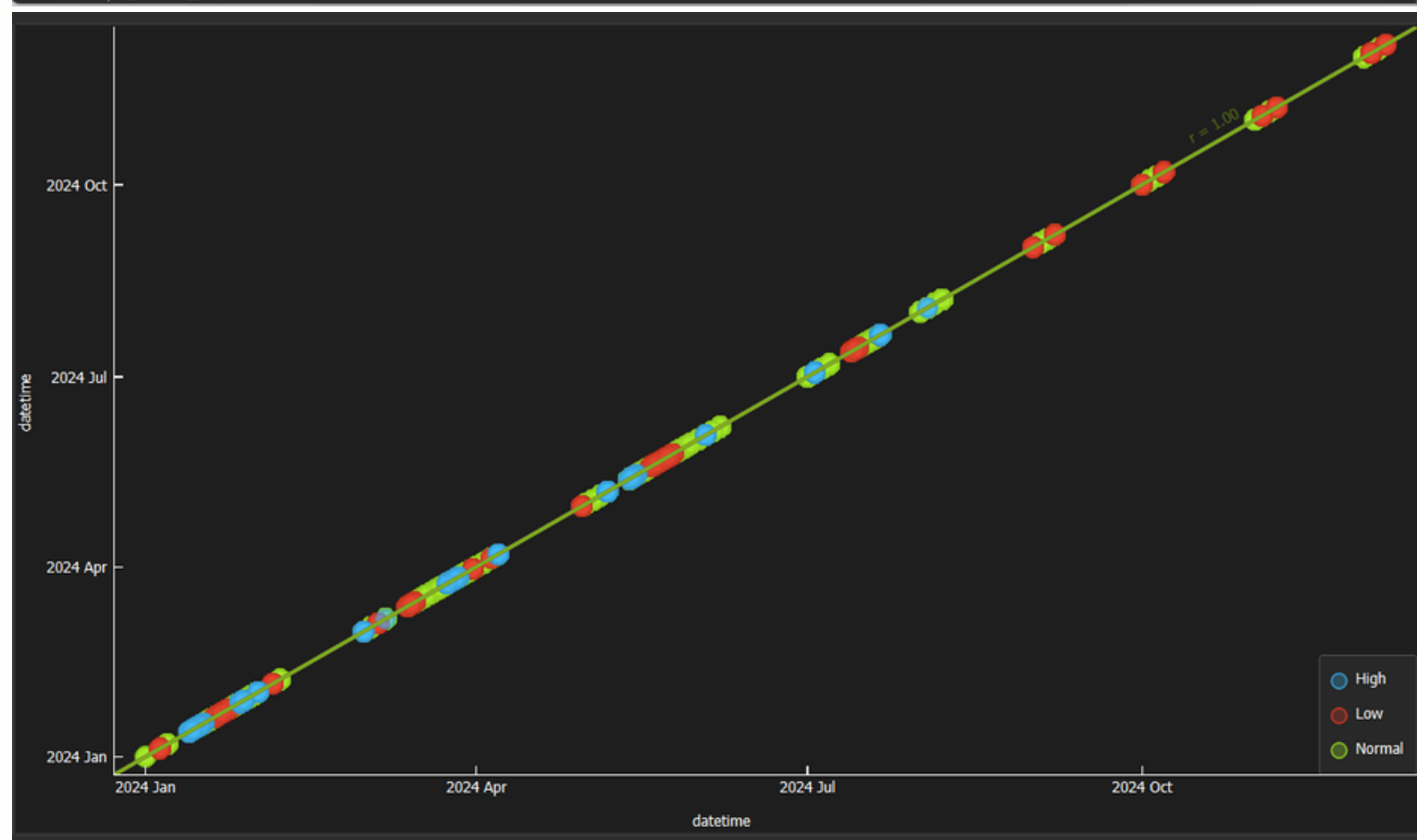
ทางกลุ่มล่องน้ำ WaterLevels
ที่ได้จากการการทำ Z-SCORE
มาเป็น Target ในการทำนาย
เพื่อให้ทราบว่า ตัวเลขที่นำเข้ามาค่าจะอยู่ในช่วงไหน



Predict Water Levels (BasicStatistics)



จะเห็นว่า Levels ของน้ำจะอยู่ในช่วงระดับ Normal เป็นส่วนใหญ่



และทางกลุ่มพบว่า ในช่วงหลังจากเดือน 9 ให้หลังจะไม่พบค่าน้ำที่สูงกว่าค่าเฉลี่ย



Test and score (WaterLevelS)

Test and Score - Orange

File Edit View Window Help

Cross validation

Number of folds: 5

☒ Stratified

Cross validation by feature

C Tree

Random sampling

Repeat train/test: 10

Training set size: 70 %

☒ Stratified

Leave one out

☐ Test on train data

☐ Test on test data

Evaluation results for target (None, show average over classes)

Model	AUC	CA	F1	Prec	Recall	MCC
Tree	1.000	1.000	1.000	1.000	1.000	1.000
Random Forest (1)	1.000	1.000	1.000	1.000	1.000	1.000
kNN	0.996	0.967	0.967	0.967	0.967	0.939
Neural Network	0.500	0.529	0.463	0.412	0.529	-0.000

Compare models by: Area under ROC curve

Negligible diff.: 0.1

	Tree	Random Forest (1)	kNN	Neural Network
Tree		0.500	0.925	1.000
Random Forest (1)	0.500		0.925	1.000
kNN	0.075	0.075		1.000
Neural Network	0.000	0.000	0.000	

Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.

2768 | - | 2768 | 4x2768

ใช้การ train model ด้วย Cross validation number of folds : 5 ครั้ง

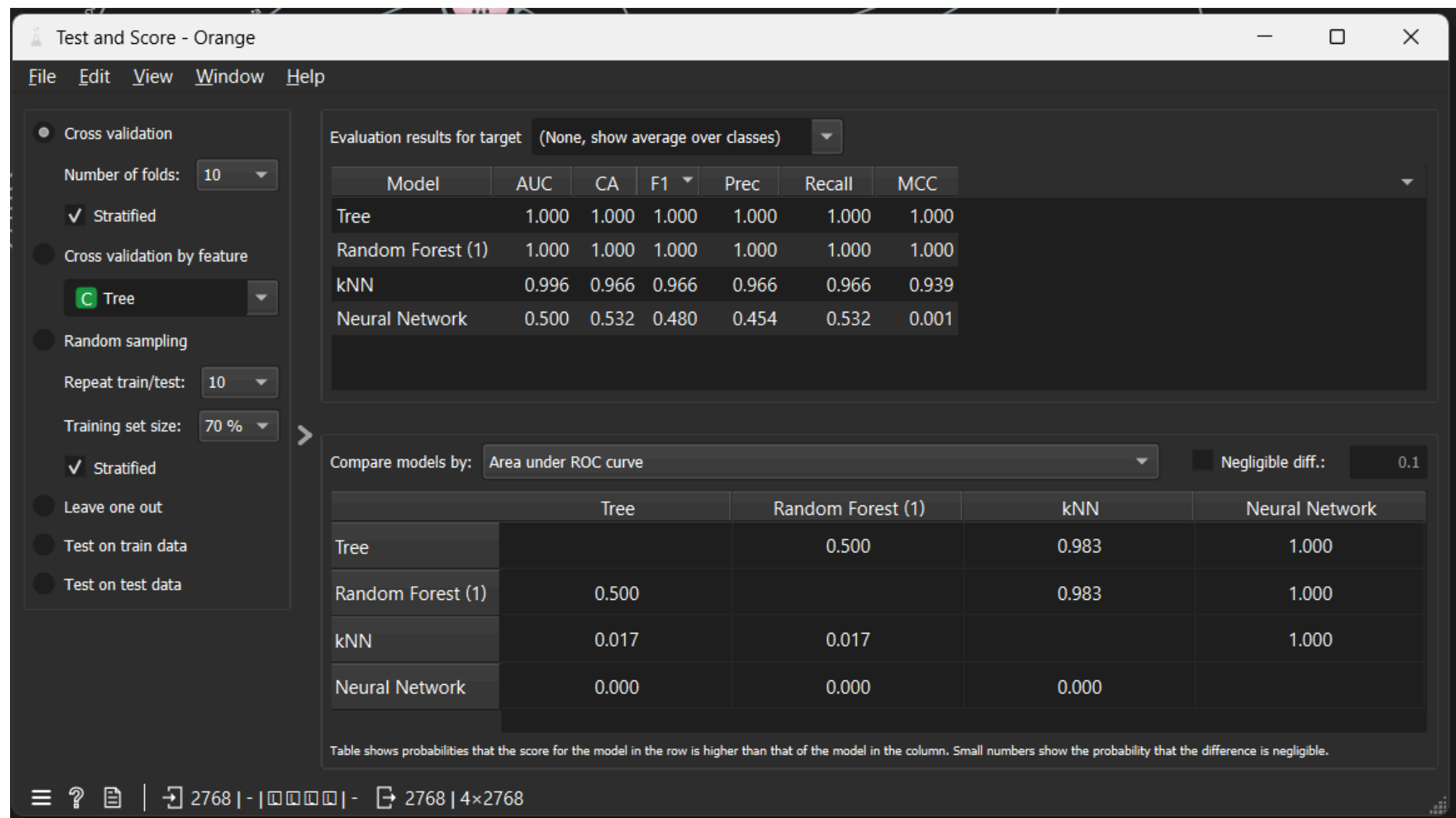
เห็นได้ว่าการ tree และ random Forest จะได้ค่าที่แม่นยำ เทียบเท่ากัน

KNN และ Neural Network จะdiffกันค่อนข้าง มาก



Test and score (WaterLevelS)

ใช้การ train model ด้วย Cross validation number of folds : 10 ครั้ง



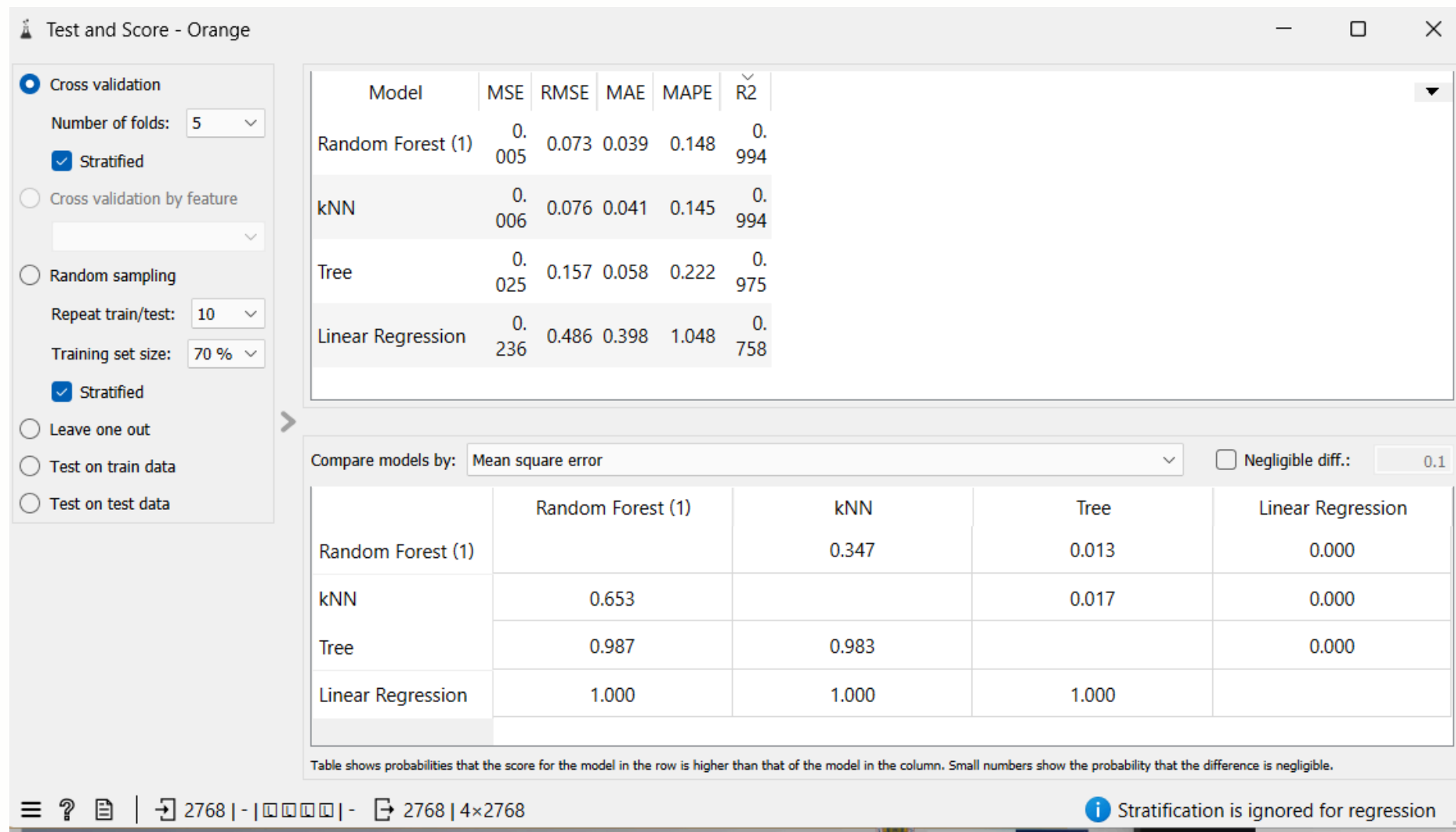
จากข้อมูลในภาพ มีโอกาสเกิด OverFitting กับ โมเดล Neural Network

- โมเดล Neural Network มีคะแนน AUC, CA, F1, Precision, Recall และ MCC บน ชุดข้อมูล Train สูงมาก (1.000)

- โมเดล Tree มีคะแนน AUC, CA, F1, Precision, Recall และ MCC ใกล้เคียงกับ Random Forest (1) แต่ Random Forest (1) มีประสิทธิภาพการทำงานที่ดีกว่าเล็กน้อย



Test and score(Values)



ใช้การ train model ด้วย Cross validation number of folds : 5 ครั้ง

เลือก Random Forest จะพบว่า การใช้ model Random Forest จะมีค่าที่ Diff กันไม่มากเมื่อเทียบกับค่า kNN Model Tree และ Linear Regression มีโอกาสที่เกิด Over Fiting มากที่สุด

- โมเดล Linear Regression มีคะแนน R² บนชุดข้อมูล Train สูงมาก (0.758)

Test and score(Values)

Test and Score - Orange

☒ Cross validation
 Number of folds: 10
☒ Stratified
☐ Cross validation by feature
☐ Random sampling
 Repeat train/test: 10
 Training set size: 70 %
☒ Stratified

Model	MSE	RMSE	MAE	MAPE	R ²
Random Forest (1)	0.005	0.072	0.038	0.141	0.995
kNN	0.005	0.074	0.039	0.131	0.994
Tree	0.022	0.150	0.053	0.207	0.977
Linear Regression	0.236	0.486	0.398	1.047	0.758

- ใช้การ train model ด้วย Cross validation number of folds : 10 ครั้ง เลือก Random Forest
- โมเดล Random Forest มีประสิทธิภาพการทำงานที่ดีกว่าโมเดลอื่น ๆ
- โมเดล Knn จะไม่ทนต่อค่า outlier และประสิทธิภาพการทำงานปานกลางเมื่อเทียบกับ Random Forest



- กลุ่มของเราเลือกใช้ Model Random Forest ด้วยเหตุผลคือมีคะแนน AUC, CA, F1, Precision, Recall, และ MCC สูงสุดสำหรับการทำนายระดับน้ำ และค่า MSE, RMSE, MAE, MAPE, และ R2 ดีที่สุดสำหรับการทำนายค่า ทั้งหมดถูกเทรนด้วยวิธี cross validation โดยใช้ Folds = 10 เพราะ Folds = 20 มีค่าที่แตกต่างน้อย