

RapidMiner Lab Quick Guide

2nd edition

DATA SCIENCE PATHWAY

Course 2: Data Analytics & Big Data
Course 3: Practical Data Analytics Using RapidMiner

Lab	Page
1. Lab: Prediction model (Lab) Part 1	1
2. Lab: Prediction model (Lab) Part 2	5
3. Lab: RapidMiner Overview Lab Small Adult Data Set	9
4. Lab: Classification Lab Pima Indians Diabetes Dataset	11
5. Lab: Regression Lab Boston House Price Dataset	17
6. Lab: Clustering Lab Titanic	25
7. Lab: Association Rule Lab Online Retail V2	30

Course 2: Data Analytics & Big Data**1. Lab: Prediction model (Lab) Part 1**

1.) การตั้งค่าตัวแปร

Data Editor						
Row No.	ID (integer) <i>id</i>	IsFail (binominal) <i>label</i>	GPAX (real) <i>regular</i>	Gender (binominal) <i>regular</i>	Department (polynomial) <i>regular</i>	Attend Score (real) <i>regular</i>
1	101	0	2.950	M	Multimedia	9
2	102	1	3.600	F	Multimedia	7
3	103	?	3.380	M	Multimedia	9.500
4	104	0	3.050	M	Multimedia	6.500
5	105	1	?	F	Multimedia	9.500
6	106	0	3.540	F	Multimedia	9.500
7	107	0	2.090	F	Computer	9.500
8	108	0	3.070	M	Computer	?
9	109	0	2.870	M	Computer	9

2.) กำหนด Filter และตั้งค่า Parameter

Create Filters: filters

Create Filters: filters
Defines the list of filters to apply.

IsFail

is not missing

Match all

Match any

☒ Preselect comparators

Add Entry

OK

Cancel

Parameters

Filter Examples

filters

Add Filters...

condition class

custom_filters

☐ invert filter

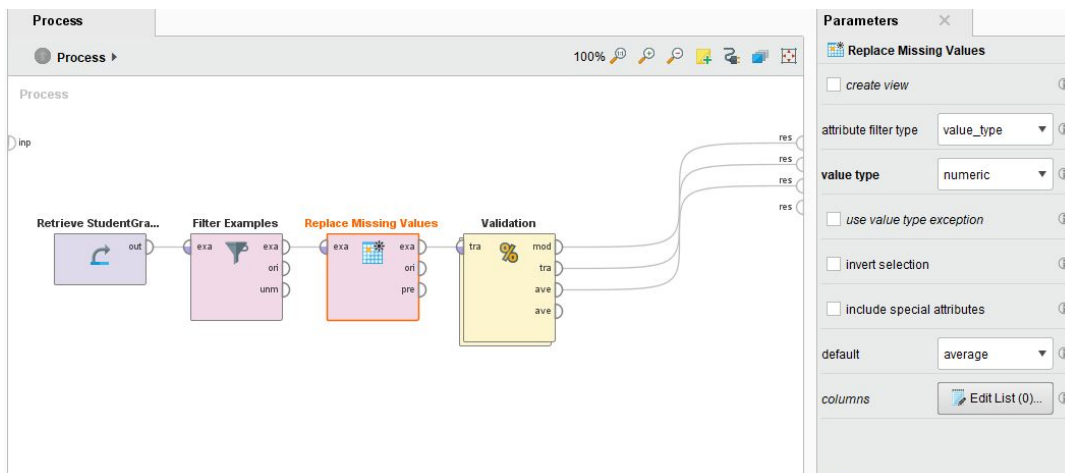
Hide advanced parameters

Change compatibility (9.0.003)

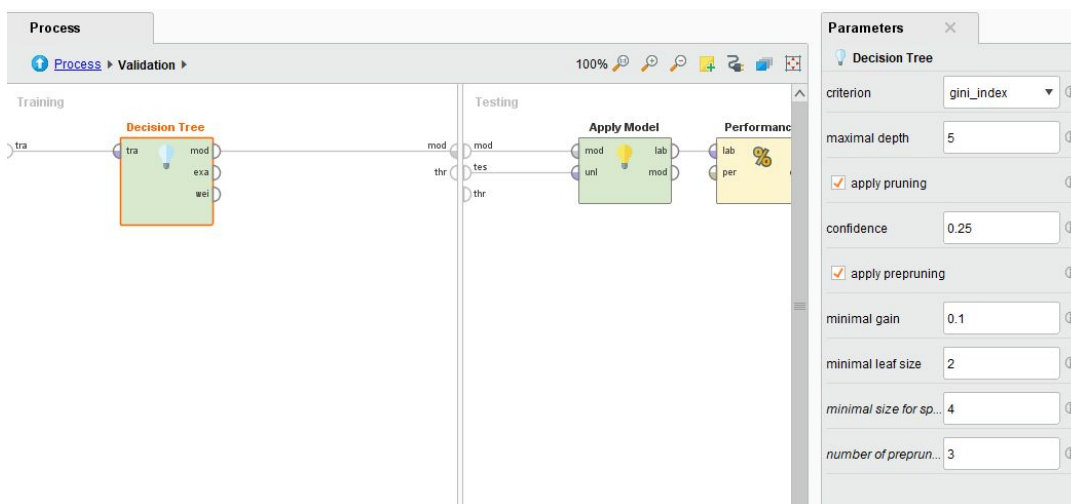
Help

Filter Examples
RapidMiner Studio Core

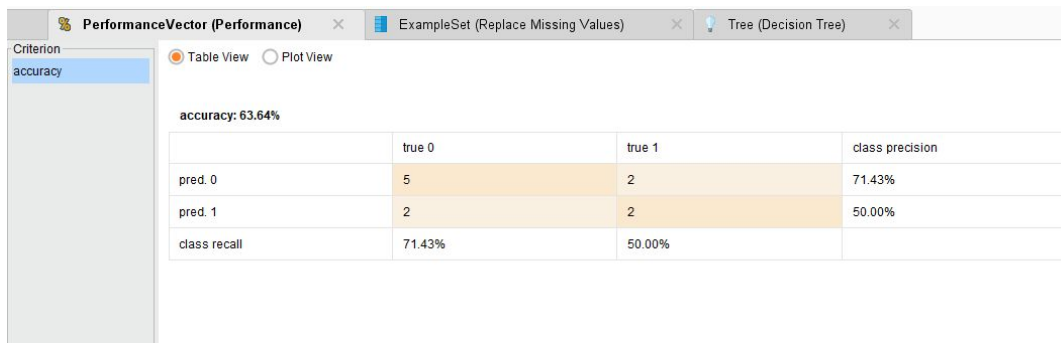
3.) Replace Missing value ตั้งค่าตามนี้



4.) การตั้งค่า Decision tree



5.) ผลจากการรันโมเดล



PerformanceVector (Performance) | ExampleSet (Replace Missing Values) | Tree (Decision Tree)

Table View | Plot View

Criterion: accuracy

accuracy: 63.64%

	true 0	true 1	class precision
pred. 0	5	2	71.43%
pred. 1	2	2	50.00%
class recall	71.43%	50.00%	

คำอธิบายเพิ่มเติมประกอบ **Lab: Prediction model (Lab) Part 1**

1. ลักษณะ Input

Input หรือ ข้อมูลนำเข้า จะมีลักษณะที่มีตัวแปรพิเศษ (Special Attributes) ได้แก่ ตัวแปรที่มีบทบาทเป็น id (ชื่อตัวแปร ID) และตัวแปรที่มีบทบาทเป็น label (ชื่อตัวแปร isFail) นอกนั้นตัวแปรอื่นๆ จะยกให้เป็นตัวแปรต้น หากพิจารณาให้เข้าใจแลบนี้จริงๆ จะต้องเข้าใจถึงการเลือกชนิดตัวแปรและชนิดบทบาทของตัวแปรนั้นๆ ว่าแต่ละตัวแปรสมควรที่จะตั้งประเภทของตัวแปรเป็นอย่างไร ยกตัวอย่าง ถ้าดูจากตัวแปร GPAX ก็คือ ตัวแปรเกรดเฉลี่ย (ค่าอยู่ระหว่างช่วง 2 ไปจนถึง 3.980) เราควรตั้งค่าตัวแปรนี้ให้อยู่ในประเภท Real Number ไม่ใช่ประเภท Integer Number เพราะข้อมูลที่เก็บมาในตัวแปรนี้เป็นค่าทศนิยม หรือค่าจำนวนจริง นั่นเอง สาเหตุที่ต้องให้ลองวิเคราะห์การตั้งค่าของตัวแปรเหล่านี้ เราจะต้องเป็นผู้พิจารณาการตั้งค่าเองนะครับ

2. Output ที่ได้

จากโจทย์ข้อนี้จะเห็นว่าเป็นการทำ Data Preparation ก่อนนั่นเอง ด้วยการเลือกตัวแปรผลเฉลยที่ต้องไม่มีค่า Missing Value ปรากฏ ด้วย Operator ที่ชื่อ Filter Example นั้นเอง สาเหตุเพราะว่า การทำงานประเภท Classification หัวใจที่สำคัญ คือ ตัวแปรผลเฉลย หรือ ตัวแปรที่ถูกกำหนดบทบาทเป็น Label เป็นตัวแปรตาม ดังนั้น การที่ข้อมูลภายในตัวแปรนี้ปรากฏ Missing เราก็ควรจะกำจัด (หรือ) แทนที่ค่า (หรือ) ไม่เลือก ค่า Missing Value โดยแลบนี้ เราเลือกที่จะ --ไม่เลือก-- ข้อมูลภายในตัวแปร Label ที่เป็น Missing มาใช้งาน สำหรับตัวแปรนำเข้าอื่นๆ เราจะใช้การแทนที่โดยค่าเฉลี่ยหากเจอ Missing ปรากฏ

เสร็จสิ้นกระบวนการทำ Data Preparation เราจะใช้นำข้อมูลที่สมบูรณ์ (ข้อมูลที่ทุก row จากทุกคอลัมน์ ไม่มีค่า Missing) ไปแบ่งข้อมูลออกเป็นสองส่วน ได้แก่ส่วนที่จะนำไปทำการ Training และส่วนที่จะนำไปใช้ในการ Testing ด้วย Operator ที่ชื่อ Split Validation จะสังเกตว่าทุกแลบจะมีการตั้งค่า Random Seed หลายคนอาจจะสงสัย

(คำถามที่ 1) ทำไมต้องตั้งค่า Random Seed ?

(คำถามที่ 2) ทำไมต้องเป็นเลข 1992 ?

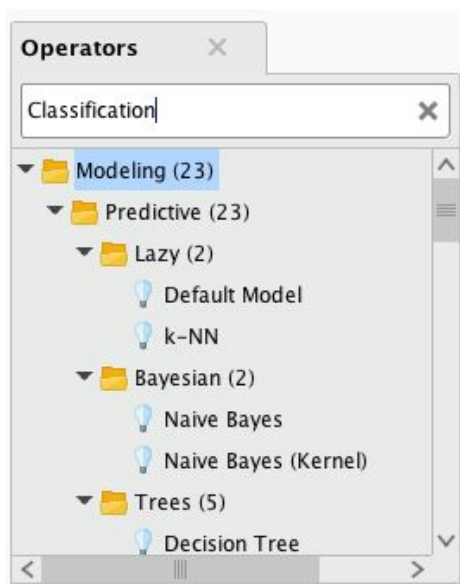
จากคำถามแรก คำตอบคือ การตั้งค่า Seed เป็นการการันตีว่าทุกรอบที่เรารันโปรแกรม จะได้ผลลัพธ์ที่เท่าเดิมตลอด เนื่องจากการ initial ค่าเริ่มต้นทั้งหลายที่เป็นแบบสุ่ม เราจะสุ่มให้มันเป็นค่าเดียวตลอด ไม่เปลี่ยนค่ามันเอง และคำตอบของคำถามสอง คือ เลข 1992 เกิดจากที่ทางทีมงานเลือกเลขนี้ขึ้นมาเป็นเลขที่จะทำให้ทุกคนเลือกเป็นเลขเดียวกัน (เลขอื่นๆ ใช้ได้ไหม เช่น 1993, 1994, 1, 2, 3, 888 คำตอบคือใช้ได้เหมือนกันหมด แต่ไม่ใช่สำหรับคอร์สนี้ละครับ 555) เพราะว่า เราจะต้องการันตีให้คำตอบของทุกคนตรงกัน หากเปลี่ยนค่า Seed คำตอบที่ได้จะผิดเพี้ยนไปนิดๆ หน่อยๆ นั้นเอง หลังจากแบ่งข้อมูลเทรนและข้อมูลเมสเราก็สร้างโมเดลจากข้อมูลเทรนด้วย Operator ที่ชื่อ Decision Tree แล้วหลังจากได้โมเดล ก็นำโมเดลไป Apply กับข้อมูลทดสอบ เพื่อวัดประสิทธิภาพบนชุดข้อมูลทดสอบ

3. Use Cases

- นำไปใช้งานในลักษณะของการทำ Data Preparation เช่น การจัดการปัญหา Missing Value
- นำไปประยุกต์กับงานประเภท Classification อื่นๆ เช่น นำไปใช้กับการทำงานที่ต้องการทำนายตาม Class ที่เรากำหนด (การทำนายว่าลูกค้าคนไหนจะซื้อโปรโมชันของเรา - Class 0 คือ ซื้อ, Class 1 คือ ไม่ซื้อ) เป็นต้น

4. Limitations & Tips

- การทำงานประเภท Classification มี Operators ให้เลือกใช้งานเพื่อสร้างโมเดล Classification ที่หลากหลาย สามารถลอง Operators ต่างๆ มาใช้สร้างโมเดลได้จาก การพิมพ์คำว่า Classification ลงไปในช่อง Search ของ Operators Box ตามรูป
- (นอกเหนือจาก Decision Tree ที่ได้ลองใช้ในแลปนี้)



2. Lab: Prediction model (Lab) Part 2

1.) การตั้งค่าตัวแปร

Data Editor						
Row No.	ID (integer) id	IsFail (binominal) label	GPAX (real) regular	Gender (binominal) regular	Department (polynomial) regular	Attend Score (real) regular
1	101	0	2.950	M	Multimedia	9
2	102	1	3.600	F	Multimedia	7
3	103	?	3.380	M	Multimedia	9.500
4	104	0	3.050	M	Multimedia	6.500
5	105	1	?	F	Multimedia	9.500
6	106	0	3.540	F	Multimedia	9.500
7	107	0	2.090	F	Computer	9.500
8	108	0	3.070	M	Computer	?
9	109	0	2.870	M	Computer	9

2.) กำหนด Filter และตั้งค่า Parameter

Create Filters: filters

Create Filters: filters

Defines the list of filters to apply.

IsFail

is not missing

Match all

Match any

Preselect comparators

Add Entry

OK

Cancel

Parameters

Filter Examples

filters

Add Filters...

condition class

custom_filters

invert filter

Hide advanced parameters

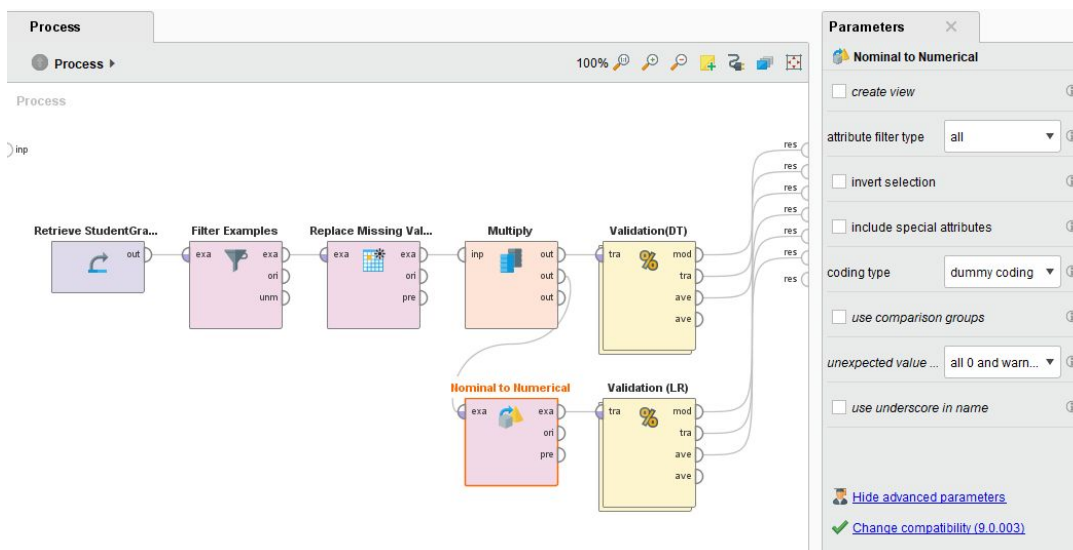
Change compatibility (9.0.003)

Help

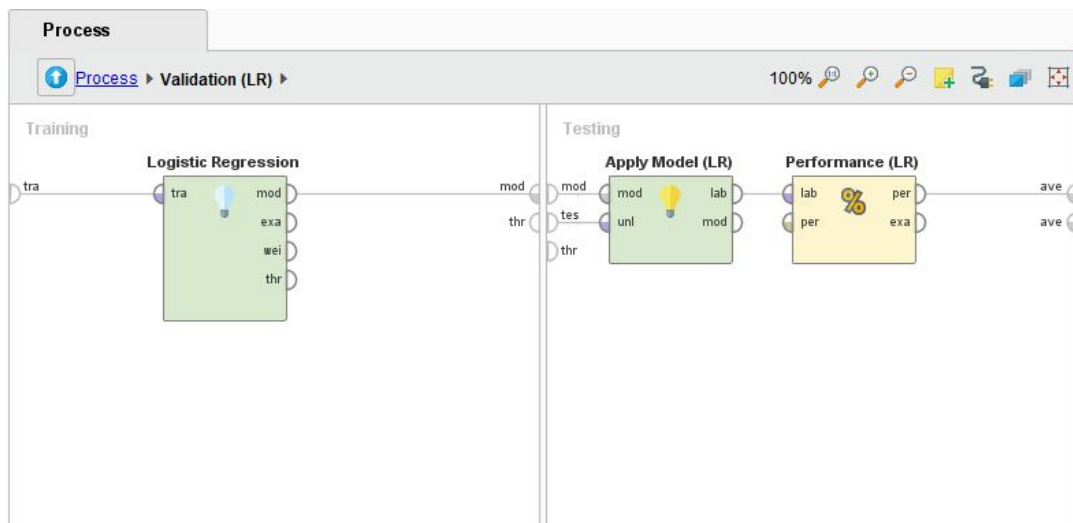
Filter Examples

RapidMiner Studio Core

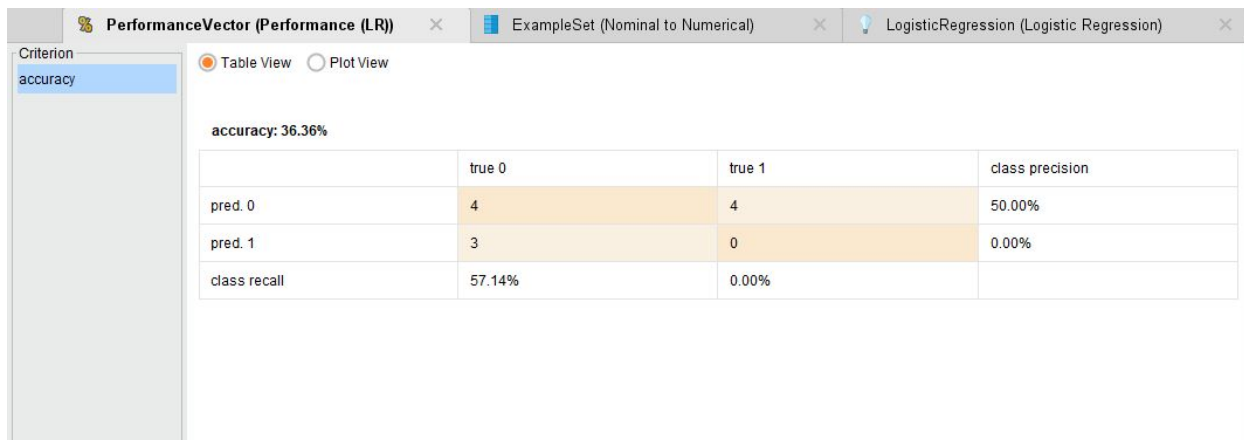
3.) การตั้งค่า Nominal to Numerical



4.) การตั้งค่าด้านใน Process



5.) ผลของการรันโมเดล



The screenshot shows the PerformanceVector window for a Logistic Regression model. The 'accuracy' criterion is selected, showing a value of 36.36%. Below this is a confusion matrix table.

	true 0	true 1	class precision
pred. 0	4	4	50.00%
pred. 1	3	0	0.00%
class recall	57.14%	0.00%	

คำอธิบายเพิ่มเติมประกอบ **Lab: Prediction model (Lab) Part 2**

1. ลักษณะ Input

- ลักษณะของ Input จะเหมือนกับที่ได้กล่าวไปจาก **Lab: Prediction model (Lab) Part 1**

2. Output ที่ได้

- ลักษณะของ Output จะเหมือนกับที่ได้กล่าวไปจาก **Lab: Prediction model (Lab) Part**
- จะต่างกับที่ Performance ของแต่ละโมเดล โดยเพิ่ม Operator ที่ชื่อ Logistic Regression เข้ามาสร้างโมเดลเพิ่มอีกหนึ่งโมเดล จะสังเกตว่าไหนดประเภท Regression, หรือ Nural Networks จะต้องมีการแปลงตัวแปรประเภทข้อความ (Nominal) ให้กลายเป็นตัวแปรประเภทตัวเลข (Numerical) เพื่อให้คำนวณเกิดเป็นสมการแบ่งคลาสได้ (Decision Boundary) ดังนั้น จึงต้องแปลงโดยการใช้ Operator ที่ชื่อ Nominal to Numerical นั้นเอง
 - คำถามต่อมา -- เราจะแปลง Nominal ให้เป็น Numerical ด้วยวิธีการใดดี (Coding Type) ?
วิธีเลือก Coding Type ให้ดูลักษณะของตัวแปร String (ตัวแปรที่เป็นข้อความ) ว่ามีลักษณะอย่างไรนะครับ เช่น ข้อมูลภายในตัวแปร ไม่สามารถเปรียบเทียบกันได้ ยกตัวอย่าง ข้อมูลจังหวัด เช่น กทม, นนทบุรี, ชลบุรี จะเห็นได้ว่าข้อมูลเหล่านี้ หากแปลงเป็น unique integer นั้นหมายความว่าเราจะให้

$$\text{กทม} = 1 \quad \text{นนทบุรี} = 2 \quad \text{ชลบุรี} = 3$$

ซึ่งทำให้ความหมายผิด กล่าวคือ ชลบุรี จะมีค่ามากกว่า นนทบุรี ? (ก็ไม่ควรเป็นเช่นนั้น) ดังนั้น เราจึงนิยมใช้การทำ Dummy Code เข้ามาแปลงนั่นเอง จะได้

กทม = 001 นนทบุรี. = 010 ชลบุรี = 100 มองเป็นตำแหน่ง แทนนะครับ

ข้อเสีย ของ Dummy Code คือ ถ้าภายในตัวแปรนั้น ข้อมูลค่อนข้างที่จะไม่ unique ความกว้างของ Bit ก็จะเยอะตามข้อมูลในตัวแปรนั้นๆ อย่างไรก็ตาม ข้อมูลภายในตัวแปร สามารถเปรียบเทียบกันได้ ยกตัวอย่าง ข้อมูลเหรียญรางวัล เช่น เหรียญทอง เหรียญเงิน เหรียญทองแดง ก็ใช้ unique interger ได้เลยครับ จะได้ เหรียญทองแดง = 1 เหรียญเงิน = 2 เหรียญทอง = 3 อันนี้ จะ make sense มากๆ เพราะ เหรียญทอง ย่อมมีค่า มากกว่า เหรียญเงิน อยู่แล้ว (กรณี เปรียบเฉพาะ เหรียญการแข่งขันทั่วๆไป นะครับ) ส่วนการตั้งค่าอื่นๆ อันนี้ต้องลองหา อ่านเพิ่มเติม แต่จะไม่หลุดออกไปจาก 2 หลักการข้างต้นนะครับ

3. Use Cases

- นำไปประยุกต์ใช้กับงานที่มีทั้งตัวแปรประเภท Nominal และประเภท Numeric ได้ เช่น วิเคราะห์งานในธนาคาร (ตัวอย่าง การทำ Fraud Detection)
- สามารถวิเคราะห์เปรียบเทียบหลายๆโมเดล จาก 1 ชุดข้อมูลได้

4. Limitations & Tips

- ถ้าใช้โมเดลประเภท Regression ถ้าชุดข้อมูลเรามีตัวแปรประเภทข้อความ จะต้องมีการใช้ โหนด Nominal to Numerical เสมอ

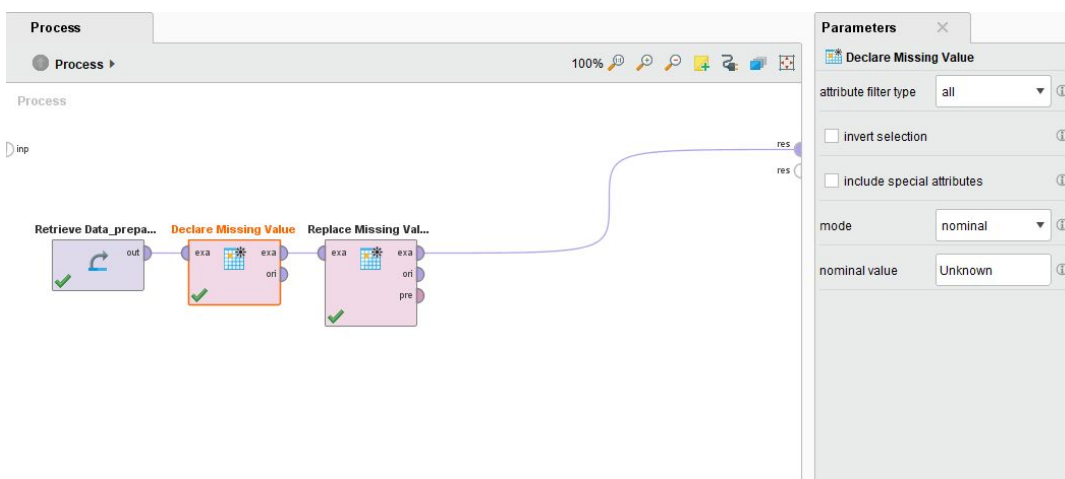
Course 3: Practical Data Analytics Using RapidMiner

3. Lab: RapidMiner Overview Lab | Small Adult Data Set

1.) ตั้งค่า ID และ Label

Name	Type	Missing	Statistics			Filter (12 / 12 attributes): <input type="text" value="Search for Attributes"/>
Id citizen-id	Integer	0	Min 1001	Max 4000	Average 2500.500	
Label salary-class	Polynomial	0	Least >50K (734)	Most <=50K (2266)	Values <=50K (2266), >50K (734)	
age	Integer	0	Min 17	Max 90	Average 38.822	
workclass	Polynomial	0	Least Unknown (0)	Most Private (2254)	Values Private (2254), Self-emp-not-inc	
education	Polynomial	0	Least Preschool (4)	Most HS-grad (970)	Values HS-grad (970), Some-college (65)	
education-num	Integer	0	Min 1	Max 16	Average 10.080	

2.) การตั้งค่า Declare Missing value



3.) ผลการรับ Process

ExampleSet (3000 examples, 2 special attributes, 10 regular attributes)

Filter (3,000 / 3,000 examples): all

Row No.	citizen-id	salary-class	age	workclass	education	education-n...	marital-status	occupation	race	sex
1	1001	<=50K	39	State-gov	Bachelors	13	Never-married	Adm-clerical	White	Male
2	1002	<=50K	50	Self-emp-not...	Bachelors	13	Married-civ-s...	Exec-manag...	White	Male
3	1003	<=50K	38	Private	HS-grad	9	Divorced	Handlers-cle...	White	Male
4	1004	<=50K	53	Private	11th	7	Married-civ-s...	Handlers-cle...	Black	Male
5	1005	<=50K	28	Private	Bachelors	13	Married-civ-s...	Prof-specialty	Black	Femal
6	1006	<=50K	37	Private	Masters	14	Married-civ-s...	Exec-manag...	White	Femal
7	1007	<=50K	49	Private	9th	5	Married-spou...	Other-service	Black	Femal
8	1008	>50K	52	Self-emp-not...	HS-grad	9	Married-civ-s...	Exec-manag...	White	Male
9	1009	>50K	31	Private	Masters	14	Never-married	Prof-specialty	White	Femal
10	1010	>50K	42	Private	Bachelors	13	Married-civ-s...	Exec-manag...	White	Male
11	1011	>50K	37	Private	Some-college	10	Married-civ-s...	Exec-manag...	Black	Male
12	1012	>50K	30	State-gov	Bachelors	13	Married-civ-s...	Prof-specialty	Asian-Pac-Isl...	Male
13	1013	<=50K	23	Private	Bachelors	13	Never-married	Adm-clerical	White	Femal

คำอธิบายเพิ่มเติมประกอบ Lab: RapidMiner Overview Lab | Small Adult Data Set

1. ลักษณะ Input

- สำหรับแลปนี้เราต้องกำหนดชนิดและบทบาทของตัวแปรเอง หลังจากเรียนเนื้อหาภายในคอร์สมาแล้ว โดยจากโจทย์ เราต้องวิเคราะห์เองให้ได้ก่อนว่า อะไรควรเป็นตัวแปรต้น และอะไรควรเป็นตัวแปรเป้าหมาย (Label) ดังนั้น จากการวิเคราะห์จะพบว่า เราควรจะทำนายระดับของเงินเดือน (salary-class) สำหรับโจทย์นี้ น่าจะเหมาะสมที่สุด ดังนั้น เราจึงกำหนดตัวแปร salary-class ให้เป็นตัวแปร Label จากนั้น เราจะสังเกตว่า มีตัวแปร citizen-id ควรจะปรับบทบาทให้เป็นชนิด id (ให้มองเหมือนมันเป็นตัวแปรที่เราได้เรียกเป็นตัวแทนของตัวอย่างข้อมูลนั้นๆ หรือ มองเป็น primary key จะสังเกตว่า ตัวอย่างของข้อมูลภายในตัวแปรประเภท id จะไม่ซ้ำกัน) เท่ากับว่า ตอนนี้เราจะมีตัวแปรประเภท Special 2 ตัวแปร (Label และ ID)
- จากนั้น ที่เหลือเราจะกำหนดให้เป็นตัวแปรต้นเพื่อสร้างโมเดล และต้องพิจารณาบทบาทชนิดตัวแปร (Polynomial, Real, Integer) จากตัวอย่างที่กล่าวไปข้างต้นจากแลปที่ผ่านมา

2. Output ที่ได้

- สำหรับแลปนี้เป้าหมายอยากให้ผู้เรียน รู้จักการตั้งค่าชนิดและบทบาทของตัวแปรด้วยตัวเอง และลองทำ Data Preparation ในเบื้องต้น เช่น การประกาศให้บางตัวอย่างกลายเป็นค่า Missing ด้วย Operator ที่ชื่อ Declare Missing Value และการแทนที่ค่า Missing Value ด้วย Operator ที่ชื่อ Replace Missing Value ดังนั้น Output ที่ได้ จะได้ตัวแปรที่มีข้อมูลสมบูรณ์พร้อมนำไปใช้สร้างโมเดลต่อไป
- ข้อแตกต่างระหว่าง Declare Missing Value และ Replace Missing Value
 - Declare คือ การเปลี่ยนค่าจากค่าอื่นๆ ให้กลายเป็นค่า Missing

- เช่น เปลี่ยนจากค่า 0 ให้กลายเป็นค่า Missing
- Replace คือ การเปลี่ยนค่า Missing ให้กลายเป็นค่าอื่นๆ
- เช่น เปลี่ยนจากค่า Missing ให้กลายเป็นค่า 0

3. Use Cases

- นำไปกำหนดชนิดและบทบาทของตัวแปรต่างๆ บนชุดข้อมูลใหม่ ได้อย่างถูกต้อง และสามารถรู้จักการทำ Data Preparation ในเบื้องต้นได้

4. Limitations & Tips

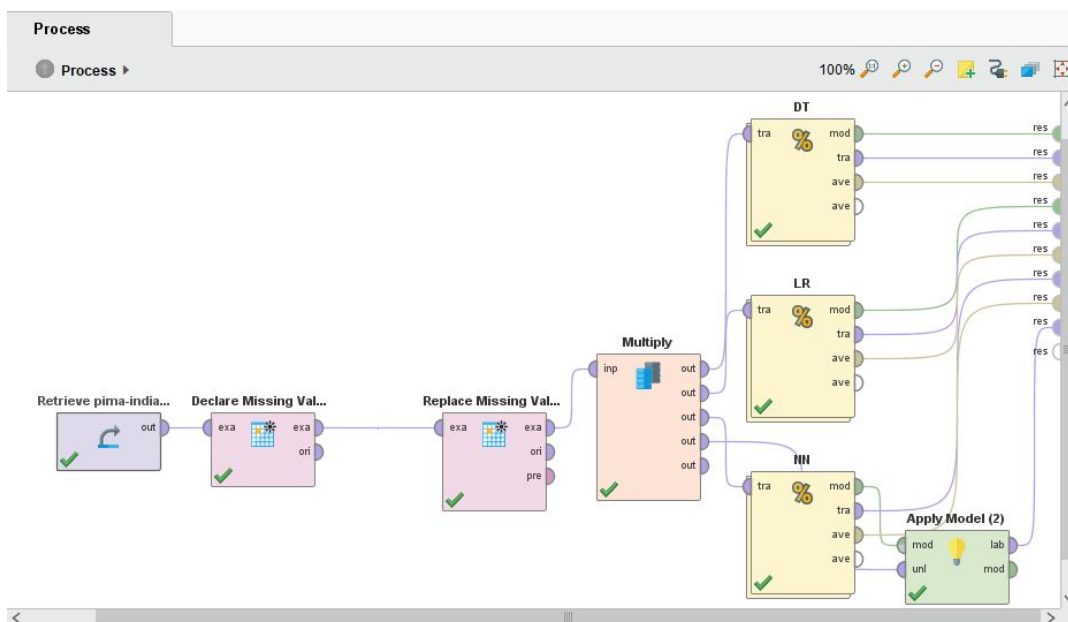
- การตั้งค่าชนิดตัวแปรมีผลกระทบต่อประสิทธิภาพของโมเดล

4. Lab: Classification Lab | Pima Indians Diabetes Dataset

1.) การตั้งค่าตัวแปร

Data Editor					
Row No.	PatientID (polynomial) id	HasDiabetes (binomial) label	NumberOfPregnant (integer) regular	Glucose (integer) regular	BloodPressure (integer) regular
1	1	1	6	148	72
2	2	0	1	85	66
3	3	1	8	183	64
4	4	0	1	89	66
5	5	1	0	137	40
6	6	0	5	116	74
7	7	1	3	78	50
8	8	0	10	115	0
9	9	1	2	197	70

2.) การตั้ง Process ตามที่โจทย์บอกเป็นดังนี้



3.) การตั้งค่า Parameter ในหน้า Declare missing value

Parameters

Declare Missing Value

attribute filter type: all

☐ invert selection

☐ include special attributes

mode: numeric

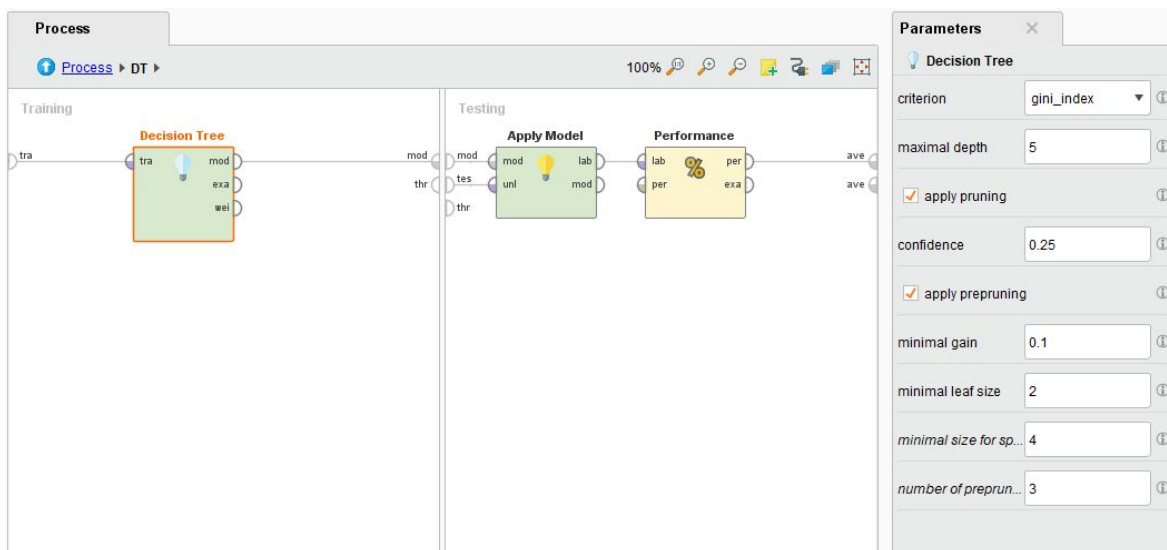
numeric value: 0.0

4.) การตั้งค่า Parameter ในหน้า Replace missing value

The screenshot shows the 'Parameters' window for the 'Replace Missing Values' operator. The window has a title bar with a close button. The main area contains several settings:

- create view**: A checkbox that is currently unchecked.
- attribute filter type**: A dropdown menu set to 'all'.
- invert selection**: A checkbox that is currently unchecked.
- include special attributes**: A checkbox that is currently unchecked.
- default**: A dropdown menu set to 'average'.
- columns**: A button labeled 'Edit List (0)...'.

5.) การตั้งค่าหน้า Process Decision tree



6.) ผลจากการรันโมเดล Decision tree เป็นดังนี้

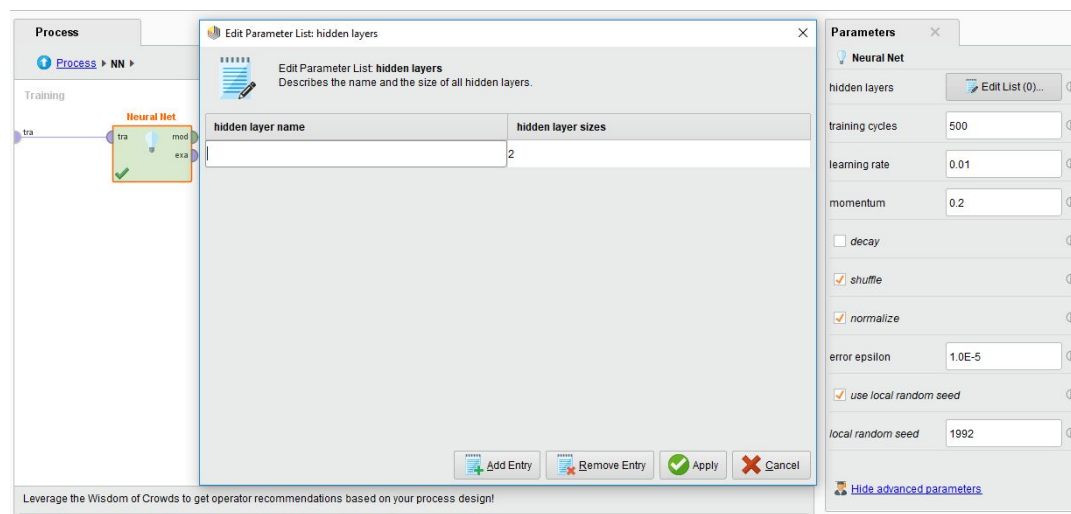
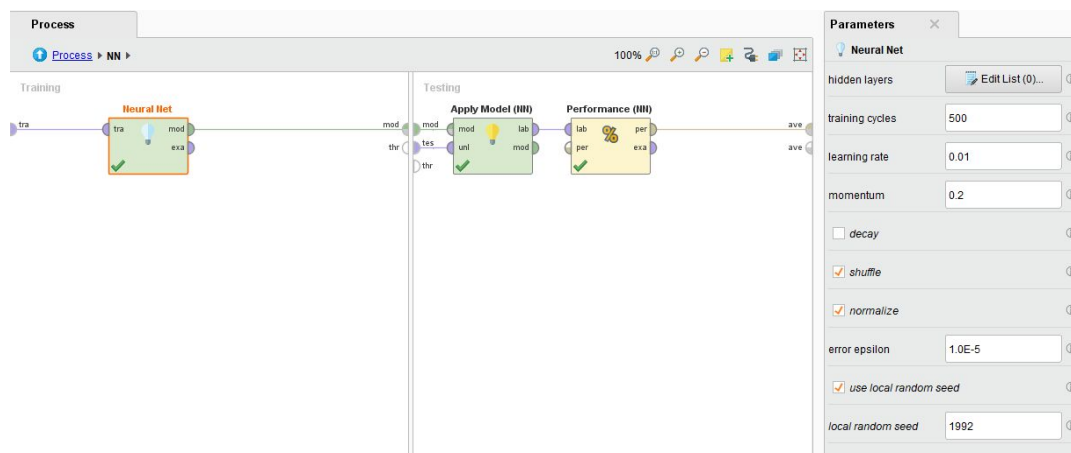
PerformanceVector (Performance) x ExampleSet (Multiply) x Tree (Decision Tree)

Table View Plot View

accuracy: 69.57%

	true 1	true 0	class precision
pred. 1	34	24	58.62%
pred. 0	46	126	73.26%
class recall	42.50%	84.00%	

7.) การตั้งค่าใน Process Neural network



8.) ผลการรันโมเดล Neural Network

Criterion
accuracy

Table View Plot View

accuracy: 74.78%

	true 1	true 0	class precision
pred. 1	43	21	67.19%
pred. 0	37	129	77.71%
class recall	53.75%	86.00%	

คำอธิบายเพิ่มเติมประกอบ Lab: Classification Lab | Pima Indians Diabetes Dataset

1. ลักษณะ Input

- แลปนี้เป็นตัวแทนของแบบฝึกหัดงานประเภท Classification ซึ่งอยู่ในหมวดของ Supervised Learning หรือ การเรียนแบบมีผู้ฝึกสอน ดังนั้น จะต้องมิตัวแปรผลเฉลยที่แน่นอน (Label Variable) หากพิจารณาในตัวแลปดีๆ จะพบว่ามิตัวแปรที่เป็นหมายเลขคนไข้ ดังนั้น จึงต้องกำหนดบทบาทให้กับตัวแปรนี้ให้เป็นประเภท id ด้วย ส่วนตัวแปรที่เหลือให้ใช้เป็นต้นแปรต้น และต้องทำ Data Preparation ด้วยการ Declare และ Replace Missing Value เพื่อให้ข้อมูลมีความสมบูรณ์ก่อนนำไปสร้างโมเดล

2. Output ที่ได้

- จะได้ผลลัพธ์จากโมเดลที่สร้างขึ้นมาเพื่อเปรียบเทียบประสิทธิภาพ 3 โมเดล จากนั้น เลือกโมเดลที่ให้ Performance (Classification) ที่ดีที่สุด ไปใช้งานต่อไป

3. Use Cases

- นำไปวิเคราะห์ของแลปนี้ไปประยุกต์ใช้กับงานประเภทที่จะต้องทำนายคลาสที่มีคำตอบรออยู่แน่ๆ เช่น อยากนำไปใช้ทำนายว่าลูกน้องจะมีแนวโน้มที่จะลาออกจากบริษัทภายในปีนี้หรือไม่ (ออก / ไม่ออก), อยากนำไปใช้ทำนายว่าปีนี้การแข่งขันเตะกร้อของโรงเรียนแห่งหนึ่งจะได้เหรียญอะไร (เหรียญทอง / เหรียญเงิน / เหรียญทองแดง) เป็นต้น

4. Limitations & Tips

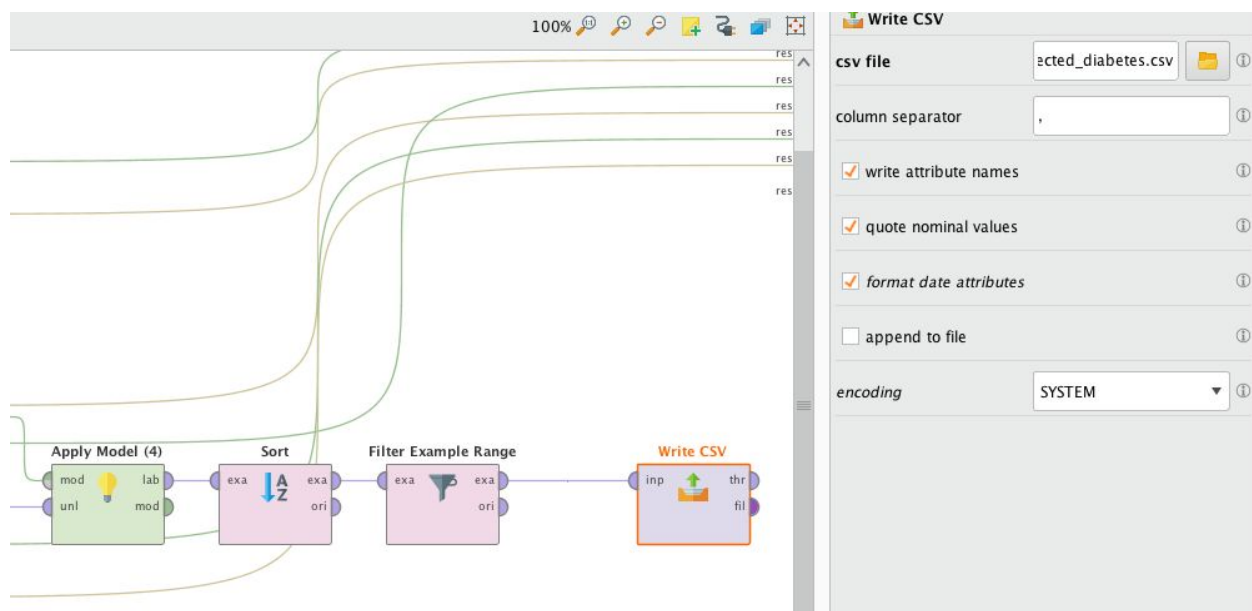
- วิธีการอ่านค่า Performance ของ Confusion Matrix แนะนำให้อ่านค่าเฉพาะคลาสที่เราสนใจจะดีกว่าอ่านค่า Performance รวมนะครับ ยกตัวอย่างในแลปนี้เลย

- ให้ดูค่าที่ทำนายเฉพาะคลาสที่เราสนใจ นั่นก็คือ คลาส 1 ดังนั้น จะเห็นได้ เราทำนายว่าคนไข้จะเป็นโรคเบาหวาน และ ผลที่เราทำนายเมื่อนำไปเทียบกับผลเฉลย แล้วทำนายถูกต้องจริงๆ มีทั้งหมด 41 ราย ดังนั้น ให้เราพิจารณาคลาส Precision และ คลาส Recall ที่ตกที่คลาส 1 จะได้ค่าเท่ากับ 65.08 % และ 51.25 % ตามลำดับ ดังนั้น ถ้าอยากจะเพิ่มประสิทธิภาพโมเดลให้ดียิ่งขึ้น ต้องทำให้ ทำนายคนไข้ให้ถูกมากกว่า 41 คนขึ้นไป แล้ว ค่า Precision และ Recall ของคลาส 1 ควรจะต้องสูงขึ้นครับ ไม่ใช่ ไปปรับโมเดลให้คลาสที่เราไม่ได้สนใจ (คลาส 0) สูงขึ้นครับ

f_measure: 80.76% (positive class: 0)

	true 1	true 0	class precision
pred. 1	41	22	65.08%
pred. 0	39	128	76.65%
class recall	51.25%	85.33%	

- หลังจากได้ค่าการทำนายคนไข้ (ยกตัวอย่างเทียบจากแลปนี้) สามารถส่งต่อการทำนายให้อยู่ในรูปแบบของ Excel เพื่อส่งให้คุณหมอพิจารณาคนไข้ที่มีโอกาสเป็นโรคเบาหวานสูงๆ ให้ดูแลคนไข้รายนี้เป็นพิเศษต่อไป (ใช้ Operator "Write CSV" ในการ Save ผลลัพธ์ไปใช้งานต่อ)



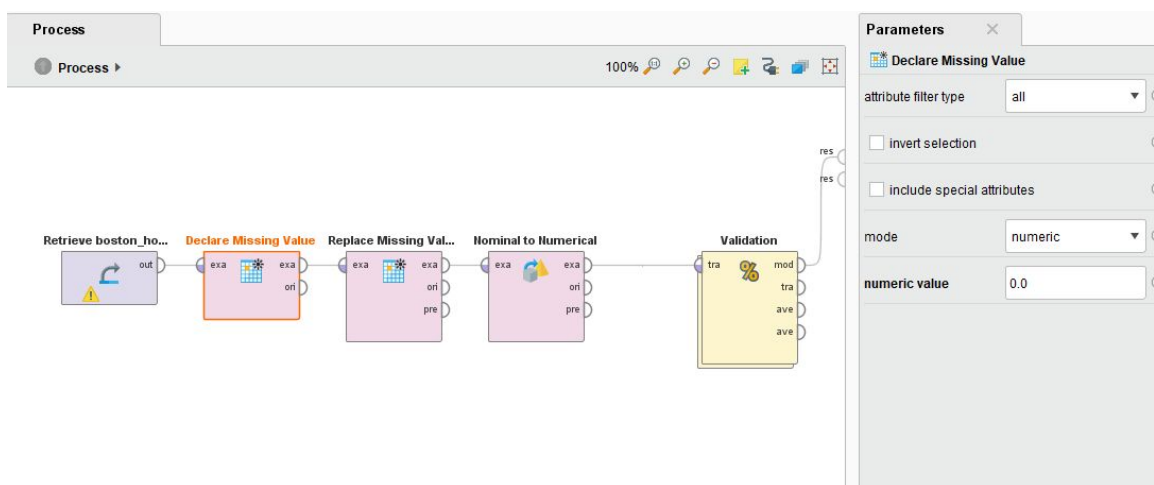
5. Lab: Regression Lab | Boston House Price Dataset

5.1) Lab1

1.) การตั้งค่าตัวแปร

Data Editor							
Row No.	PRICE (real) label	CRIM (real) regular	ZN (real) regular	INDUS (real) regular	CHAS (binominal) regular	NOX (real) regular	RM (real) regular
1	24	0.006	18	2.310	0	0.538	6.575
2	21.600	0.027	0	7.070	0	0.469	6.421
3	34.700	0.027	0	7.070	0	0.469	7.185
4	33.400	0.032	0	2.180	0	0.458	6.998
5	36.200	0.069	0	2.180	0	0.458	7.147
6	28.700	0.030	0	2.180	0	0.458	6.430
7	22.900	0.088	12.500	7.870	0	0.524	6.012
8	27.100	0.145	12.500	7.870	0	0.524	6.172
9	16.500	0.211	12.500	7.870	0	0.524	5.631
10	18.000	0.178	12.500	7.870	0	0.524	5.955

2.) การตั้งค่าหน้า Declare missing value



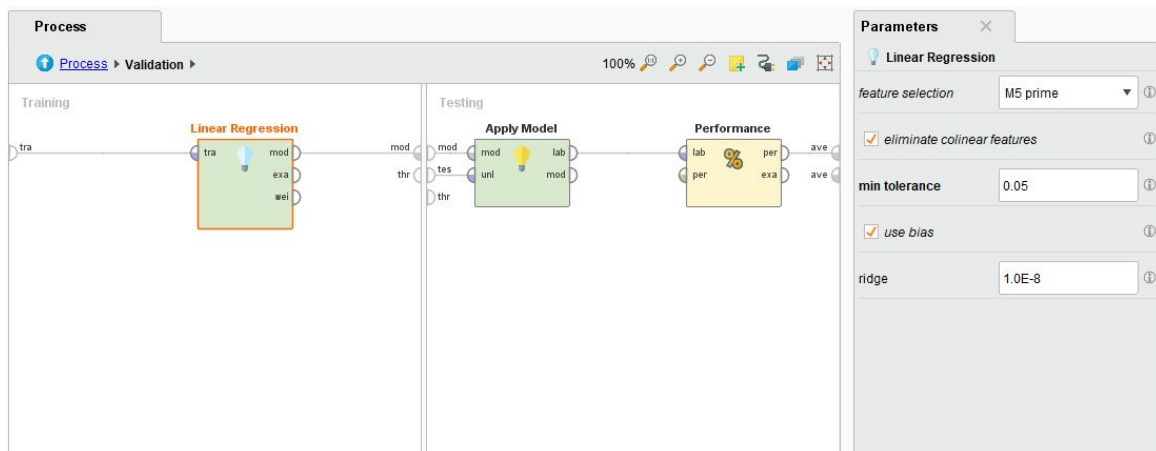
3.) การตั้งค่าหน้า Replace missing value

The screenshot shows the RapidMiner 'Process' window with a workflow consisting of five operators: 'Retrieve boston_ho...', 'Declare Missing Val...', 'Replace Missing Values', 'Nominal to Numerical', and 'Validation'. The 'Replace Missing Values' operator is highlighted in orange. To the right, the 'Parameters' panel for 'Replace Missing Values' is open, showing settings for 'attribute filter type' (all), 'invert selection' (unchecked), 'include special attributes' (checked), 'default' (average), and 'columns' (Edit List (0)...).

4.) การตั้งค่าหน้า Nominal to numerical

The screenshot shows the 'Parameters' panel for the 'Nominal to Numerical' operator. The 'coding type' is set to 'dummy coding', 'use comparison groups' is checked, and 'unexpected value han...' is set to 'all 0'. Below this, a dialog box titled 'Edit Parameter List: comparison groups' is open. It contains a table with two columns: 'comparison group attribute' and 'comparison group'. The first row has 'CHAS' in the first column and '0' in the second column. At the bottom of the dialog are buttons for 'Add Entry', 'Remove Entry', 'Apply', and 'Cancel'. Below the dialog, there is a note: 'Leverage the wisdom of crowds to get operator recommendations based on your process design!'.

5.) การตั้งค่า Process ด้านใน

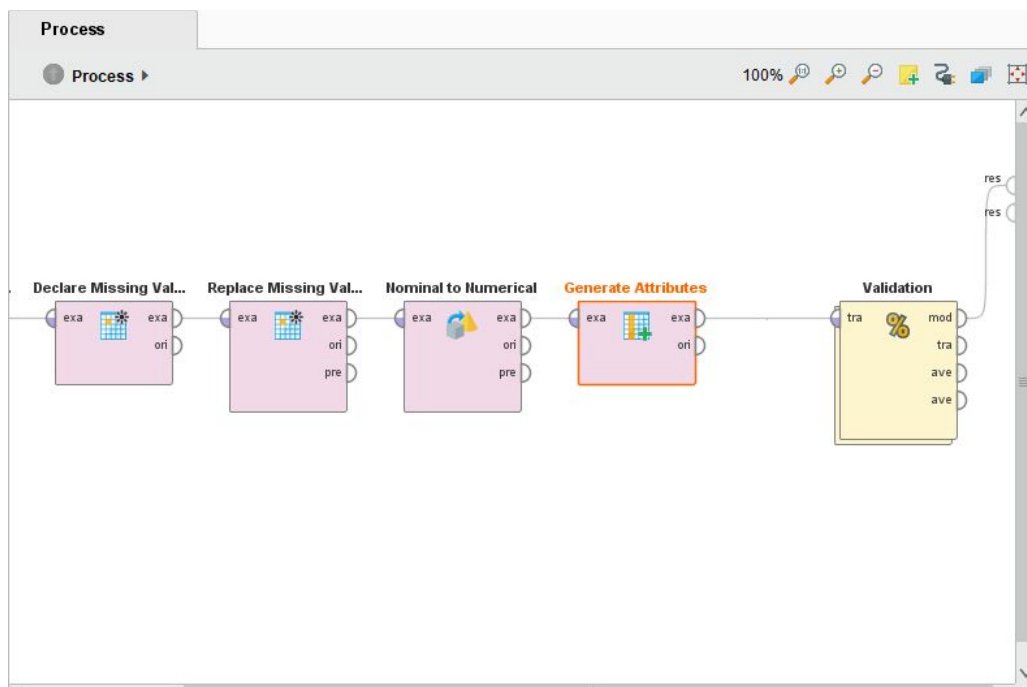


6.) ผลจากการ Run Model

LinearRegression (Linear Regression)							
Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code
CHAS = 1	5.601	1.235	0.171	1.000	4.536	0.000	****
ZN	0.089	0.024	0.162	0.991	3.642	0.000	****
INDUS	-0.548	0.079	-0.423	0.745	-6.928	0.000	****
NOX	-0.402	1.163	-0.080	0.999	-0.346	0.730	
RM	0.231	0.098	0.754	1.000	2.343	0.020	**
AGE	-0.079	0.017	-0.316	0.923	-4.560	0.000	****
DIS	-1.846	0.251	-1.436	0.965	-7.355	0.000	****
RAD	0.030	0.019	0.736	1.000	1.597	0.111	
TAX	-0.006	0.003	-0.135	0.832	-2.051	0.041	**
B	0.018	0.005	0.265	0.981	3.410	0.001	****
(Intercept)	32.635	3.332	?	?	9.796	0	****

5.2) Lab2

1.) เพิ่มกล่อง Generate Attributes และกำหนดดังภาพ



Views: **Design** Results Turbo Prep Auto Model

Find data, operators...etc 🔍 All Studio ▾

Edit Parameter List: function descriptions

Edit Parameter List: function descriptions
List of functions to generate.

attribute name	function expressions
CRIM	$\ln(\text{CRIM}+1)$
ZN	$\ln(\text{ZN}+1)$
NOX	$\ln(\text{NOX}+1)$
RM	$\ln(\text{RM}+1)$
DIS	$\ln(\text{DIS}+1)$
RAD	$\ln(\text{RAD}+1)$
TAX	$\ln(\text{TAX}+1)$

➕ Add Entry
➖ Remove Entry
✅ Apply
❌ Cancel

Parameters

Generate Attributes

function descriptions 🔍 Edit List (7)...

☒ keep all

[Hide advanced parameters](#)
[Change compatibility \(9.0.003\)](#)

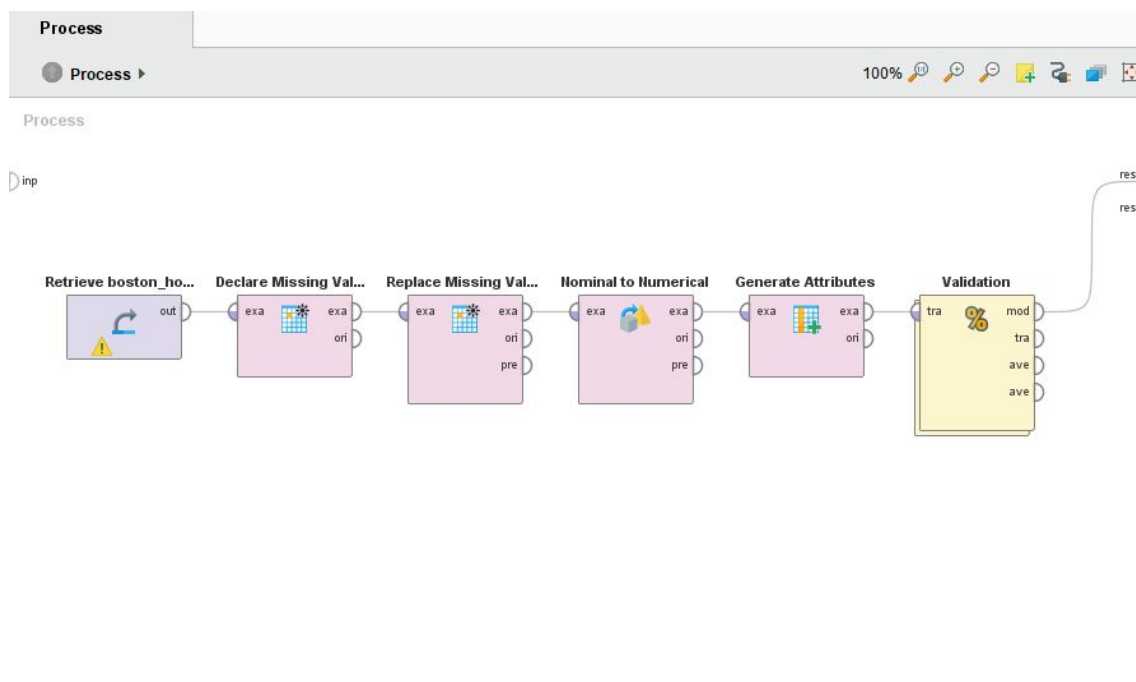
the Wisdom of Crowds to get operator recommendations based on your process design!

2.) นอกนั้นตั้งค่าเหมือนกับ Lab1 เมื่อ Run Model ออกมาจะได้ผลดังนี้

LinearRegression (Linear Regression)							
Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code
CHAS = 1	4.938	1.055	0.151	1.000	4.679	0.000	****
INDUS	-0.457	0.067	-0.353	0.829	-6.817	0.000	****
AGE	-0.087	0.015	-0.347	0.953	-5.632	0.000	****
B	0.021	0.005	0.309	0.990	4.530	0.000	****
CRIM	1.732	0.859	0.142	0.803	2.015	0.044	**
NOX	-34.656	3.580	-1.931	1.000	-9.681	0	****
RM	31.869	2.251	3.045	1.000	14.158	0	****
DIS	-13.305	1.216	-1.024	0.888	-10.941	0	****
RAD	-1.319	0.755	-0.246	0.991	-1.747	0.081	*
(Intercept)	-0.343	4.530	?	?	-0.076	0.940	

5.3) Lab3

1.) ใช้ Process เดียวกับ Lab2



2.) ในหน้า Validation ให้เปลี่ยนเป็น Neural Network และตั้งค่า Parameter ตามนี้

The screenshot shows the RapidMiner interface with the 'Validation' process selected. The process flow includes a 'Neural Net' operator in the Training section, followed by 'Apply Model' and 'Performance' operators in the Testing section. The 'Parameters' panel on the right is open for the 'Neural Net' operator, showing the following settings:

- hidden layers: Edit List (0)...
- training cycles: 500
- learning rate: 0.3
- momentum: 0.2
- ☐ decay
- ☒ shuffle
- ☒ normalize
- error epsilon: 1.0E-5
- ☐ use local random seed

3.) hidden layers ตั้งค่าตามนี้

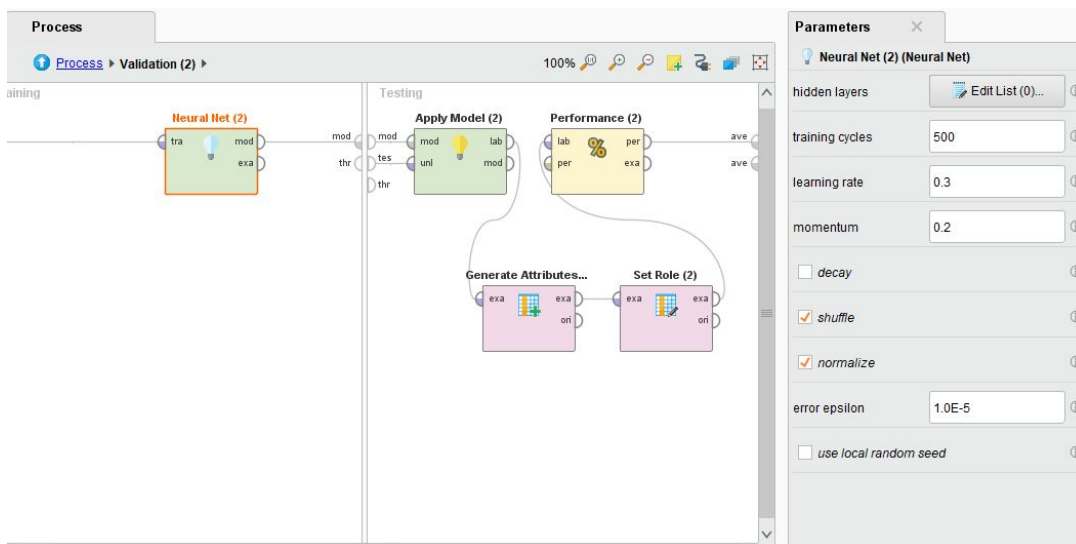
The screenshot shows the 'Edit Parameter List: hidden layers' dialog box open, allowing the user to define the hidden layers. The dialog box contains a table with the following data:

hidden layer name	hidden layer sizes
	2

At the bottom of the dialog box, there are buttons for 'Add Entry', 'Remove Entry', 'Apply', and 'Cancel'. The 'Parameters' panel on the right is also visible, showing the same settings as in the previous screenshot, but now for 'Neural Net (2) (Neural Net)'.

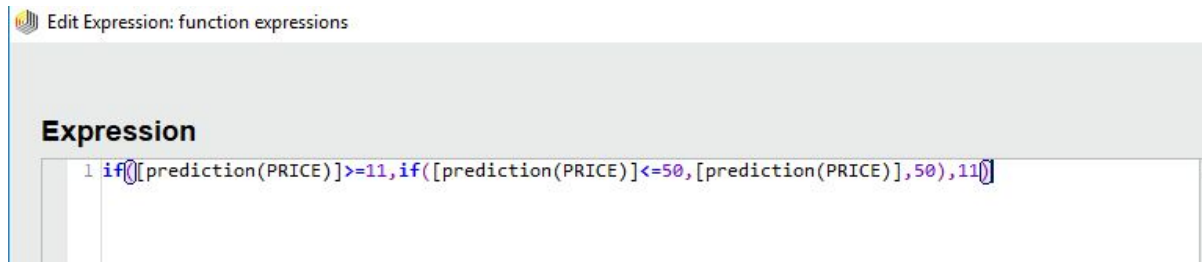
5.4) Lab4

1.) ใช้ Process เดียวกับ Lab3 แต่เพิ่มการตั้งค่าด้านใน Validation

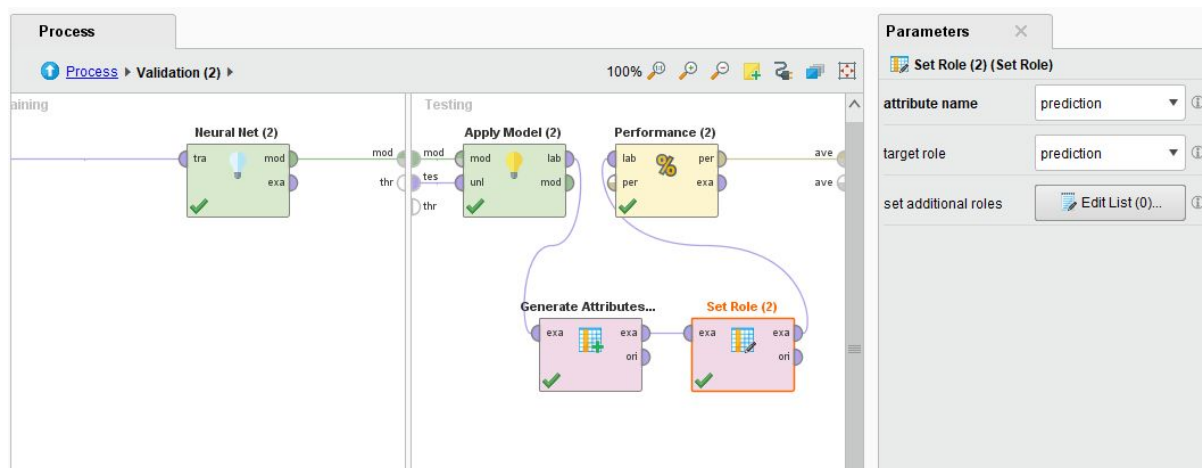


2.) การตั้งค่าใน Generate Attributes

The screenshot shows the 'Edit Parameter List: function descriptions' dialog box. The dialog has a title bar and a close button. Inside, there is a section titled 'Edit Parameter List: function descriptions' with a subtitle 'List of functions to generate.' Below this is a table with two columns: 'attribute name' and 'function expressions'. The 'attribute name' column contains the text 'prediction'. The 'function expressions' column contains the text 'If([prediction(PRICE)]>=11,if([prediction(PRICE)]<=50,[...])'. At the bottom of the dialog are buttons for 'Add Entry', 'Remove Entry', 'Apply', and 'Cancel'. To the right of the dialog is the 'Parameters' panel, which shows the settings for the 'Generate Attributes (3) (Generate Attributes)' operator. The settings include 'function descriptions' (Edit List (1)...), 'keep all' (checked), and links for 'Hide advanced parameters' and 'Change compatibility (9.0.003)'.



3.) การตั้งค่าในกล่อง Set Role



4.) เมื่อ Run Model จะได้ผลดังนี้

root_mean_squared_error

root_mean_squared_error: 3.584 +/- 0.000

คำอธิบายเพิ่มเติมประกอบ Lab: Regression Lab | Boston House Price Dataset

1. ลักษณะ Input

- ตัวแปรนำเข้าของแลปนี้ ยังอยู่ในหมวดของงานประเภท Supervised Learning หรือ อีกนัยหนึ่ง คือ จะต้องเป็นตัวแปรผลเฉลยแน่ๆ นั้น ก็คือ ตัวแปร PRICE (ราคาบ้าน) แต่จะสังเกตว่าตัวแปรที่เป็นผลเฉลยของงานประเภท Regression จะเป็นค่า Real Value หรือ จำนวนจริง จึงไม่จำเป็นต้องมีจำนวนคลาสที่แน่นอนเหมือนงานประเภท Classification (แลปก่อนหน้านี้) ส่วนตัวแปรถูกใช้เป็นตัวแปรต้น

2. Output ที่ได้

- โมเดลจะทำนายค่าราคาบ้านเป็นจำนวนจริง (Real Value) ออกมาให้ โดยคำนวณ Performance (Regression) มาให้เป็นค่า Error เช่น บ้านหลังนี้ราคาจริง 10 บาท ทำนายออกมา 8 บาท ดังนั้น ตอบผิดต่ำไปจากของจริง 2 บาท ($\text{Error} = 2$) ซึ่งในงานจริงๆ ส่วนใหญ่นิยมใช้ค่า RMSE (Root Mean Square Error) ในการดูประสิทธิภาพของโมเดล ซึ่งหลักการ การอ่าน Error ก็จะคล้ายๆ กับตัวอย่างที่กล่าวไปข้างต้น

3. Use Cases

- สามารถนำไปใช้ทำนายประเภทของราคาบ้าน, ราคาที่ดิน หรือค่าประมาณต่างๆที่ อยากรู้ได้ ที่เป็นค่าจำนวนจริง เช่น อยากรู้จะประมาณค่า Oxegen ที่เราหายใจ, ประมาณการค่าฝุ่น PM2.5

4. Limitations & Tips

- ตัวแปรต้นไม่ควรนำเข้าโมเดลมากเกินไป (ขึ้นกับดุลยพินิจในแต่ละงาน)
- ควรสังเกตการกระจายตัวของข้อมูลในแต่ละตัวแปรสัณฐาน ชีวิตจะดีขึ้น ข้อมูลควรจะกระจายตัวแบบ Normal Distribution (คือ ต้องไม่กระจายตัวของข้อมูลเบ้ไปทางด้านใดด้านหนึ่ง) จะสังเกตว่าภายในแลปจะมีการ take ln (Natural Logarithms) เพื่อดึงข้อมูลที่เบ้ ให้กลายมาเป็นการกระจายแบบรูประฆังคว่ำ (Normal Distribution)

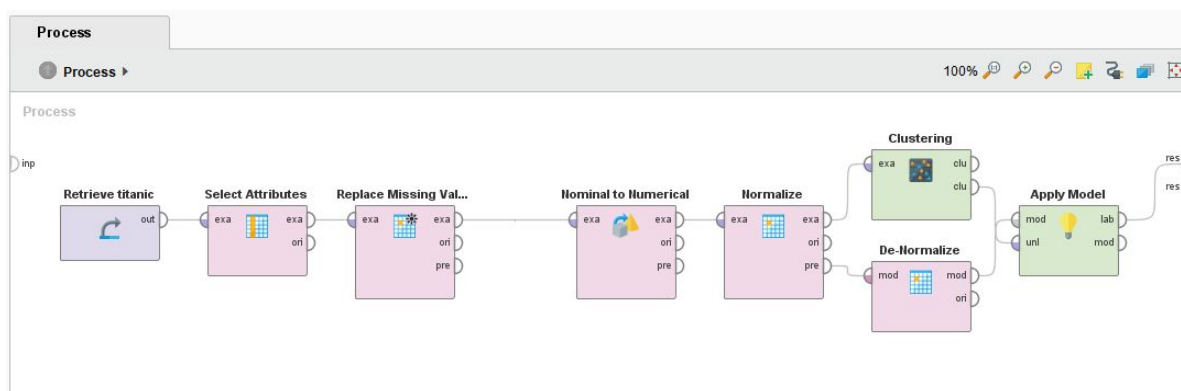
6. Lab: Clustering Lab | Titanic

1.) การกำหนดตัวแปร

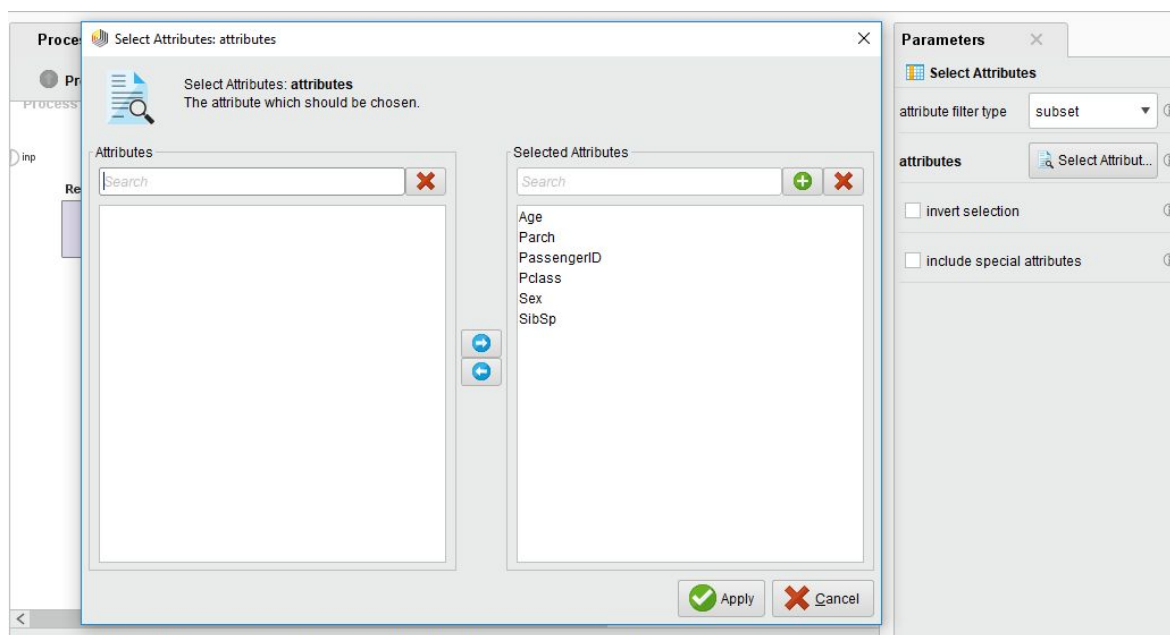
Data Editor								
Row No.	PassengerId (polynomial) id	Survived (integer) regular	Pclass (integer) regular	Name (polynomial) regular	Sex (binomial) regular	Age (real) regular	SibSp (integer) regular	Parch (integer) regular
1	1	0	3	Braund, Mr. Owen H...	male	22	1	0
2	2	1	1	Cummings, Mrs. John...	female	38	1	0
3	3	1	3	Heikkinen, Miss. Lai...	female	26	0	0
4	4	1	1	Futrelle, Mrs. Jacqu...	female	35	1	0
5	5	0	3	Allen, Mr. William H...	male	35	0	0
6	6	0	3	Moran, Mr. James	male	?	0	0
7	7	0	1	McCarthy, Mr. Timot...	male	54	0	0
8	8	0	3	Palsson, Master. Go...	male	2	3	1
9	9	1	3	Johnson, Mrs. Osca...	female	27	0	2
10	10	1	2	Nasser, Mrs. Nichol...	female	14	1	0
11	11	1	3	Sandstrom, Miss. M...	female	4	1	1

Ticket (polynomial) regular	Fare (real) regular	Cabin (polynomial) regular	Embarked (polynomial) regular
A/5 21171	7.250	?	S
PC 17599	71.283	C85	C
STON/O2. 3101282	7.925	?	S
113803	53.100	C123	S
373450	8.050	?	S
330877	8.458	?	Q
17463	51.862	E46	S
349909	21.075	?	S
347742	11.133	?	S
237736	30.071	?	C
PP 9549	16.700	G6	S
113783	26.550	C103	S

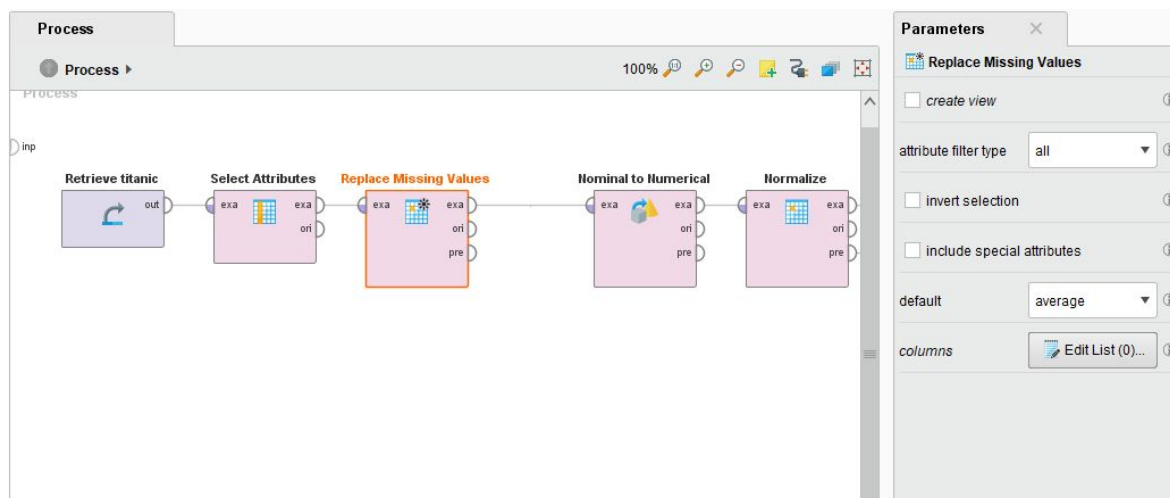
2.) Process မိန့်



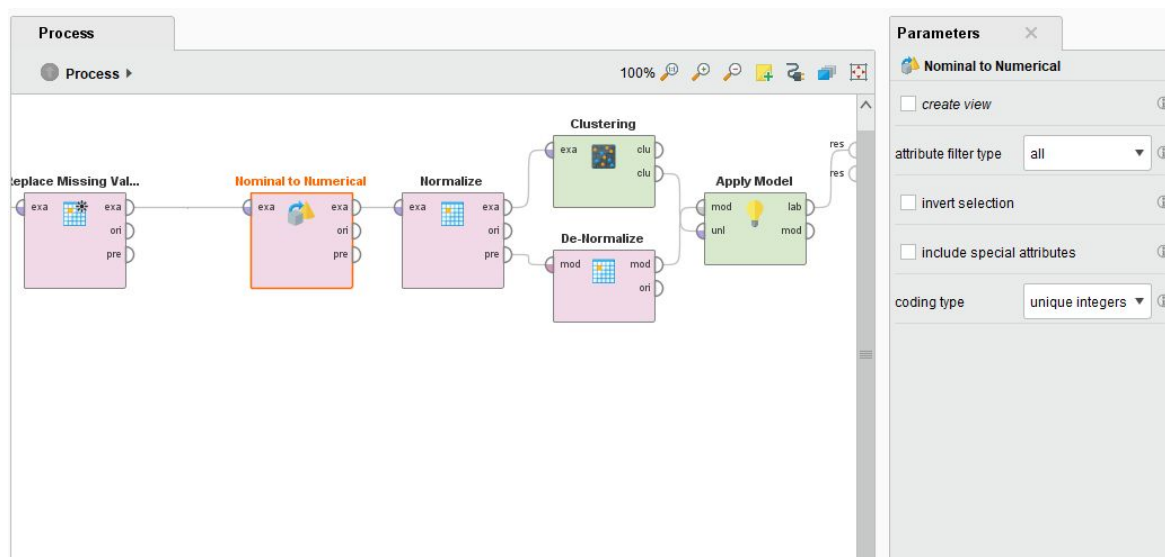
3.) การตั้งค่า Select Attributes



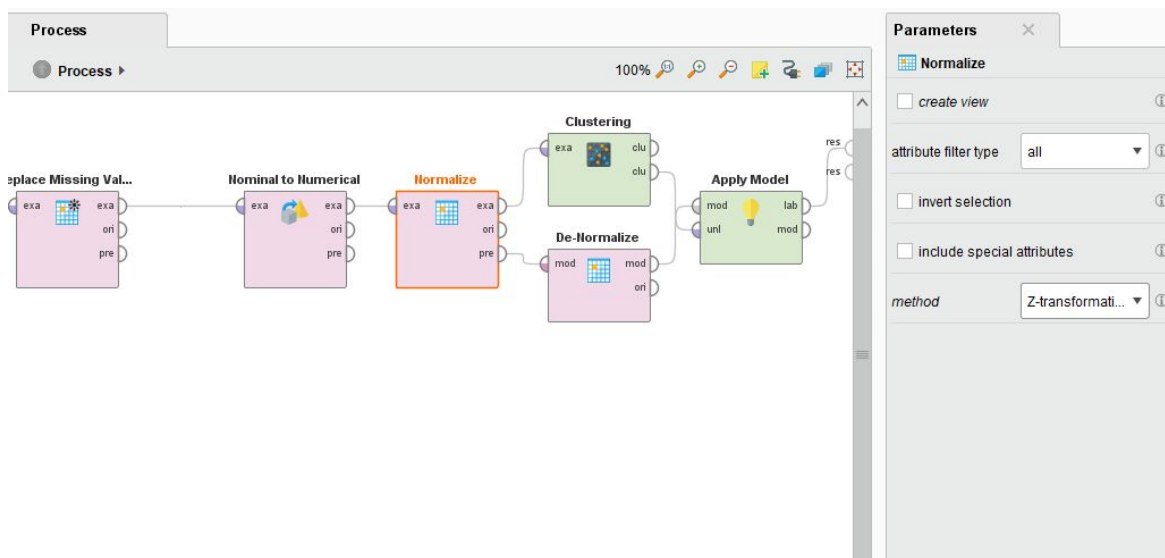
4.) การตั้งค่าในกล่อง Replace Missing Values



5.) การตั้งค่าในกล่อง Nominal to Numerical



6.) การตั้งค่าในกล่อง Normalize



7.) การตั้งค่าในกล่อง Clustering

The screenshot shows the RapidMiner process editor with a workflow: Replace Missing Values → Nominal to Numerical → Normalize → Clustering → De-Normalize → Apply Model. The Clustering process is highlighted, and its parameters are shown on the right:

- Clustering (k-Means)**
 - ☒ add cluster attribute
 - ☒ add as label
 - ☐ remove unlabeled
 - k: 3
 - max runs: 500
 - ☒ determine good start values
 - measure types: NumericalMeasures
 - numerical measure: EuclideanDistance
 - max optimization steps: 1000
 - ☒ use local random seed

Buttons at the bottom: [Hide advanced parameters](#), [Change compatibility \(9.0.003\)](#)

8.) การตั้งค่าในกล่อง De-Normalize

The screenshot shows the same workflow as above, but now the De-Normalize process is highlighted. Its parameters are shown on the right:

- De-Normalize**
 - missing attribute handling: proceed on missing

9.) จากนั้น Run Model

คำอธิบายเพิ่มเติมประกอบ **Lab: Clustering Lab | Titanic**

1. ลักษณะ Input

- สำหรับโจทย์ประเภท Clustering จะอยู่ในงานประเภท Unsupervised Learning นั้นหมายความว่า ไม่จำเป็นต้องมีตัวแปรผลเฉลย ดังนั้น เราแค่กำหนดชนิดของตัวแปรให้ถูกต้องก็เพียงพอแล้ว

2. Output ที่ได้

- จะได้ตัวแทนกลุ่มของข้อมูล ยกตัวอย่างเช่น มีตัวอย่างที่ N เข้ามาใหม่ เราสามารถบอกได้เลยว่าตัวอย่างที่ N จะถูกจัดอยู่ในกลุ่มไหน

3. Use Cases

- สามารถนำไปใช้จัดกลุ่มเพื่อเป็นตัวแทนของข้อมูลสำหรับงานที่ไม่มีผลเฉลย

4. Limitations & Tips

- พยายาม Normalize ข้อมูลทุกครั้งเมื่อทำ Clustering เพื่อให้การคำนวณ Distance ของทุกตัวแปรอยู่ในช่วงที่ใกล้เคียงกัน
- หมั่นฝึกอ่านผลจากการจัดกลุ่มทุกครั้ง กล่าวคือ ต้องรู้จักตีความผลที่ได้จากการจัดกลุ่ม เนื่องจากความยากของงานประเภท Unsupervised Learning คือ การอ่านค่าและแปลผลจากผลลัพธ์ที่ได้

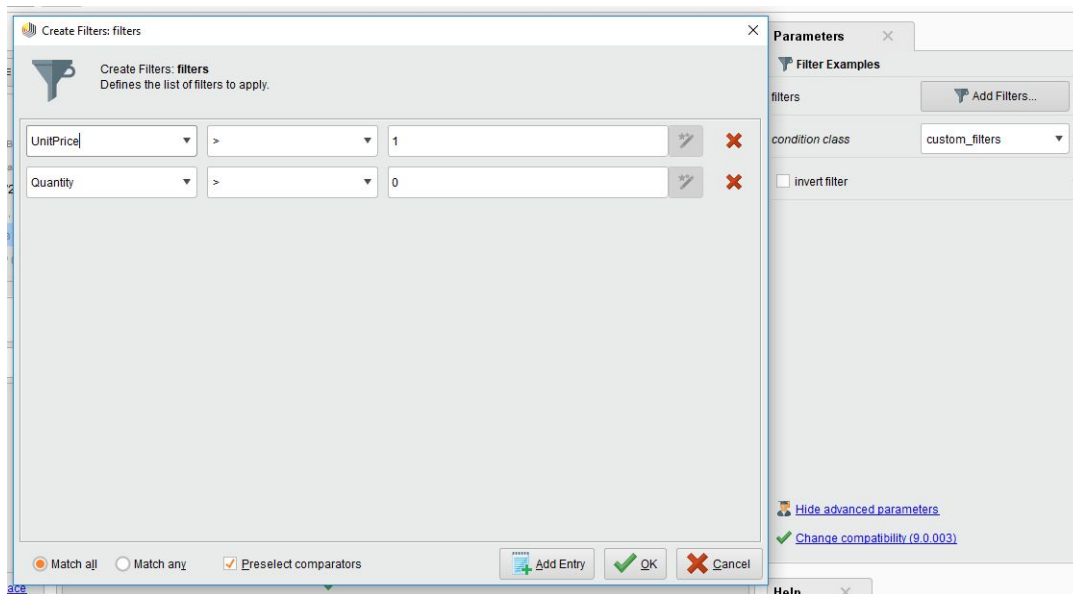
7. Lab: Association Rule Lab | Online Retail V2

1.) การกำหนดตัวแปร

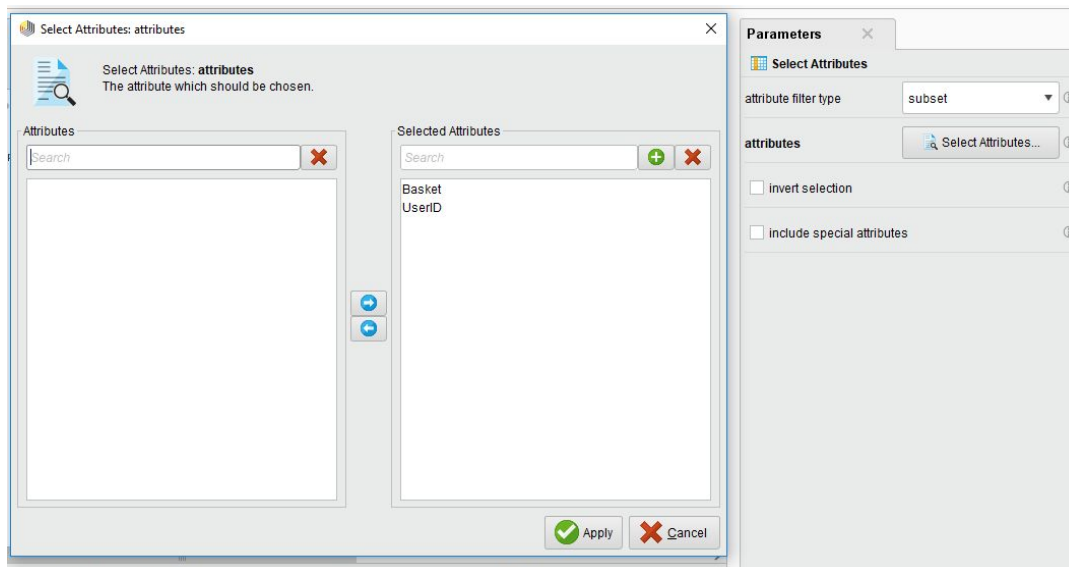
Data Editor					
Row No.	InvoiceNumber (polynomial) regular	StockCode (polynomial) regular	Basket (polynomial) regular	Quantity (integer) regular	InvoiceDate (date_time) regular
1	571729	23210	WHITE ROCKING H...	144	Oct 19, 2011 09:4.
2	571729	23211	RED ROCKING HO...	144	Oct 19, 2011 09:4.
3	571729	23212	HEART WREATH D...	144	Oct 19, 2011 09:4.
4	571729	23213	STAR WREATH DE...	144	Oct 19, 2011 09:4.
5	571729	POST	POSTAGE	1	Oct 19, 2011 09:4.
6	571730	22578	WOODEN STAR CH...	24	Oct 19, 2011 09:4.
7	571730	35970	ZINC FOLKART SLE...	12	Oct 19, 2011 09:4.
8	571730	22596	CHRISTMAS STAR ...	12	Oct 19, 2011 09:4.
9	571730	22178	VICTORIAN GLASS ...	6	Oct 19, 2011 09:4.
10	571730	23313	VINTAGE CHRISTM...	5	Oct 19, 2011 09:4.

<input type="text"/>			<input type="checkbox"/> Case sensitive
UnitPrice (real) regular	UserID (polynomial) regular	Country (polynomial) regular	
1.040	12500	Germany	
1.040	12500	Germany	
1.040	12500	Germany	
1.040	12500	Germany	
18	12500	Germany	
0.290	16851	United Kingdom	
1.690	16851	United Kingdom	
1.250	16851	United Kingdom	
1.950	16851	United Kingdom	
4.950	16851	United Kingdom	

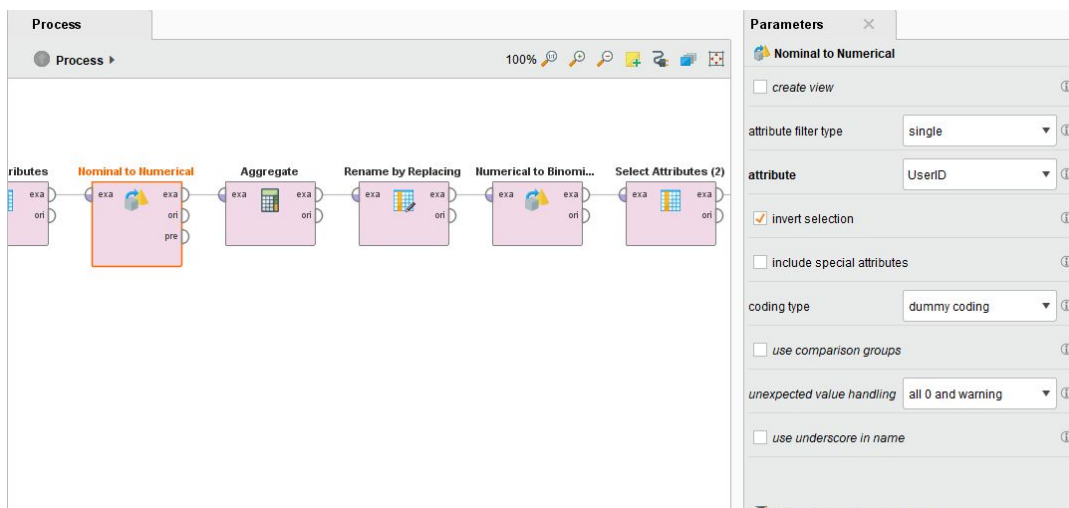
2.) กำหนดและตั้งค่า Parameter ในกล่อง Filter Examples ตามนี้



3.) การกำหนด Select Attributes



4.) การตั้งค่า Parameter ในกล่อง Nominal to Numerical



5.) การตั้งค่าในกล่อง Aggregate

The screenshot displays the RapidMiner interface. On the left, a process flow is visible with the following operators: **ributes**, **Nominal to Numerical**, **Aggregate** (highlighted in orange), **Rename by Replacing**, **Numerical to Binomi...**, and **Select Attributes (2)**. The **Aggregate** operator is selected, and its parameters are shown on the right.

Parameters

- Aggregate**
- ☒ use default aggregation
- attribute filter type: **single**
- attribute: **UserID**
- ☒ invert selection
- ☐ include special attributes
- default aggregation functi...: **maximum**
- aggregation attributes: **Edit List (0)...**
- group by attributes: **Select Attributes...**
- ☐ count all combinations
- ☐ only distinct
- [Hide advanced parameters](#)
- [Change compatibility \(9.0.003\)](#)

Leverage the Wisdom of Crowds to get operator recommendations based on your process design!

The screenshot shows the **Select Attributes: group by attributes** dialog box. The dialog has two main sections: **Attributes** and **Selected Attributes**.

Attributes

- Search:

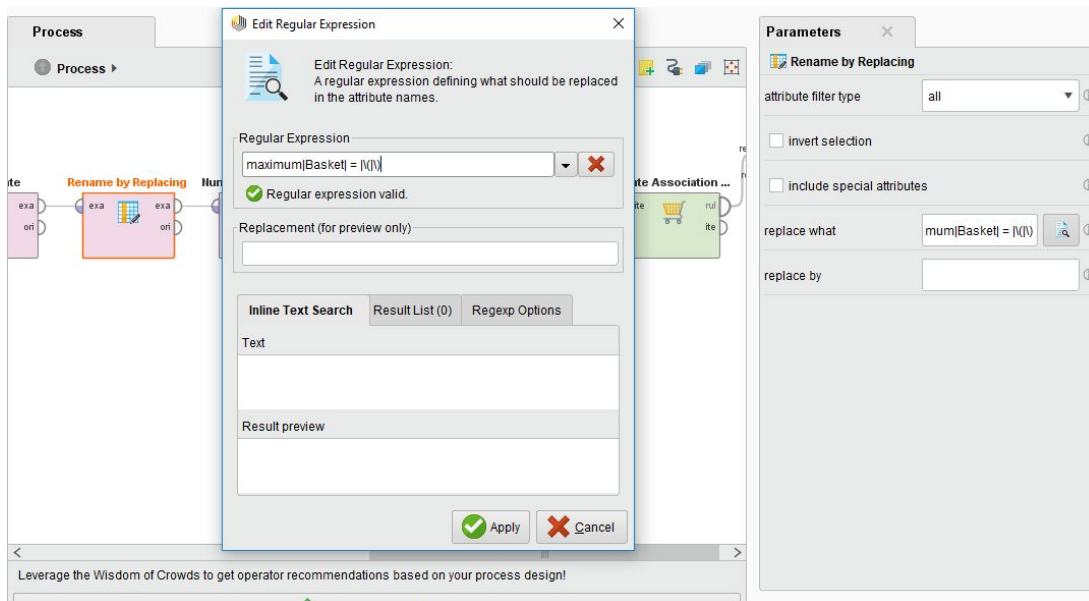
Selected Attributes

- Search:
- UserID

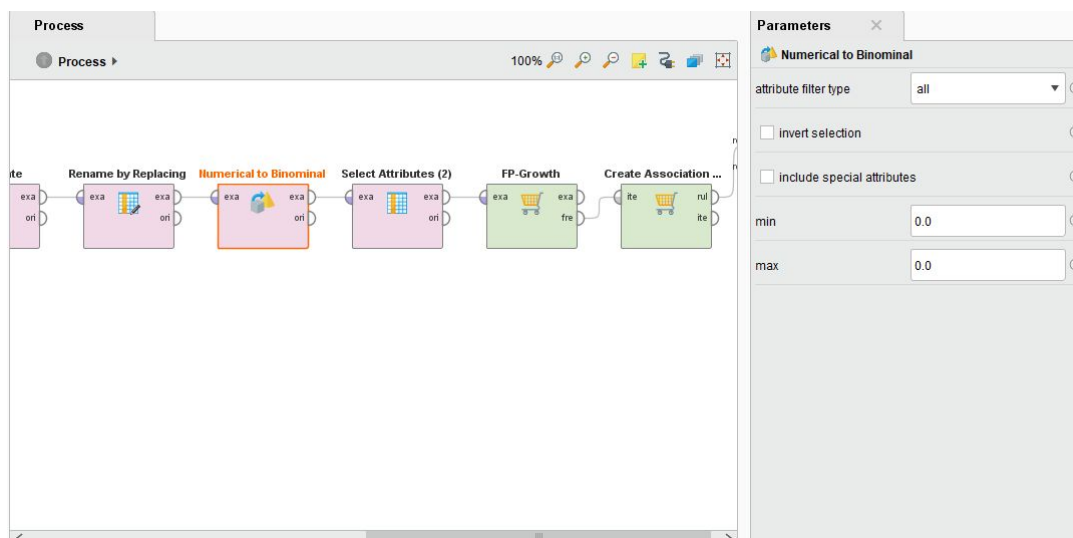
Buttons: **Apply** (green checkmark), **Cancel** (red X).

Leverage the Wisdom of Crowds to get operator recommendations based on your process design!

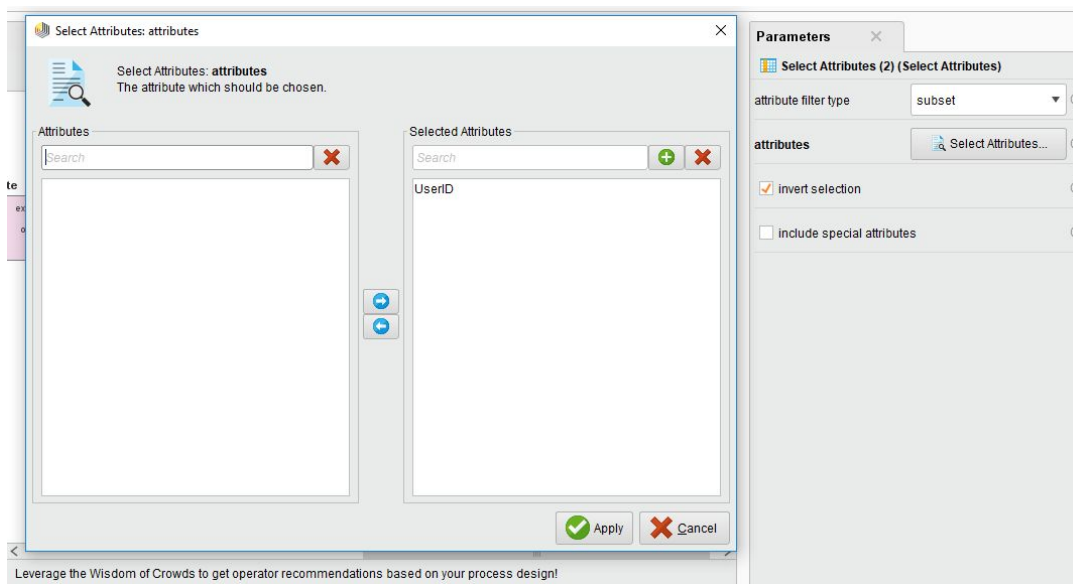
6.) การกำหนด Parameter ในกล่อง Rename by Replacing



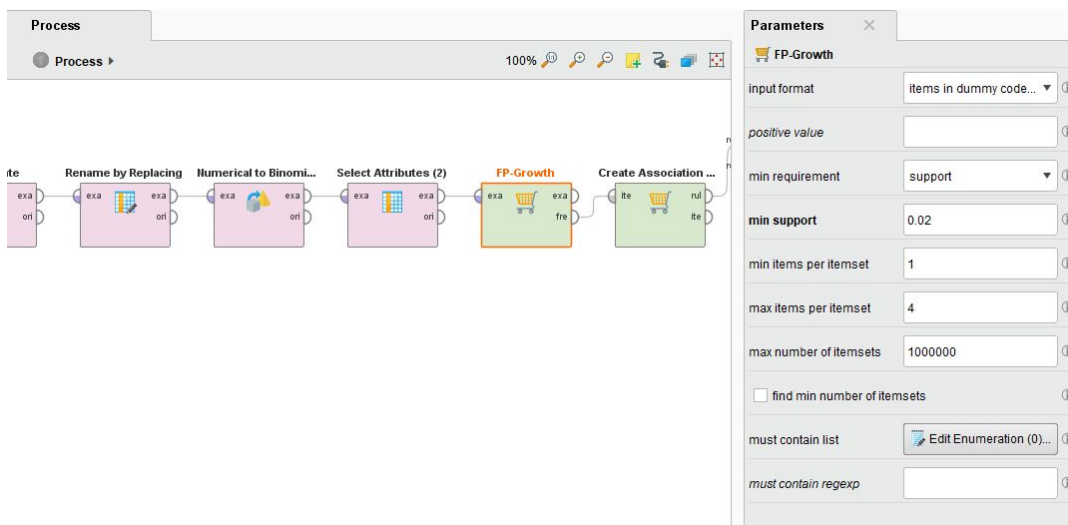
7.) การตั้งค่า Parameter ในกล่อง Numerical to Binominal



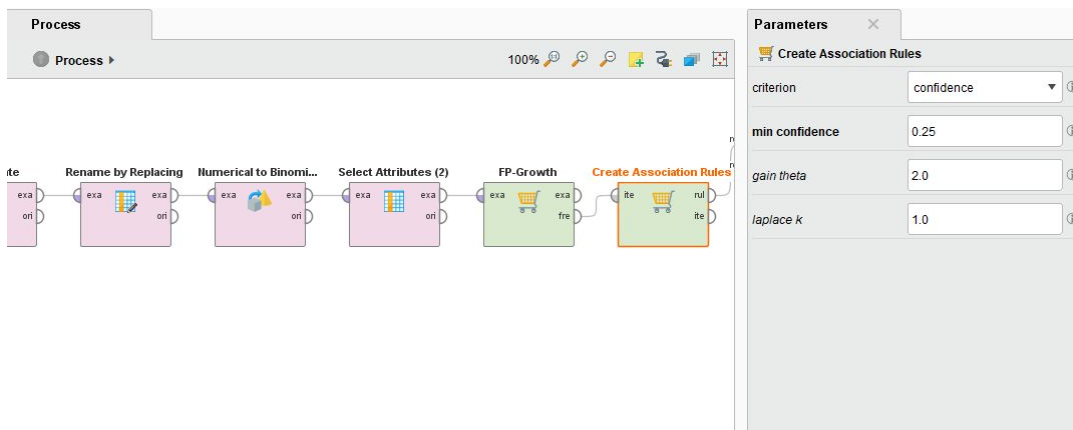
8.) การตั้งค่าในกล่อง Select Attributes(2)



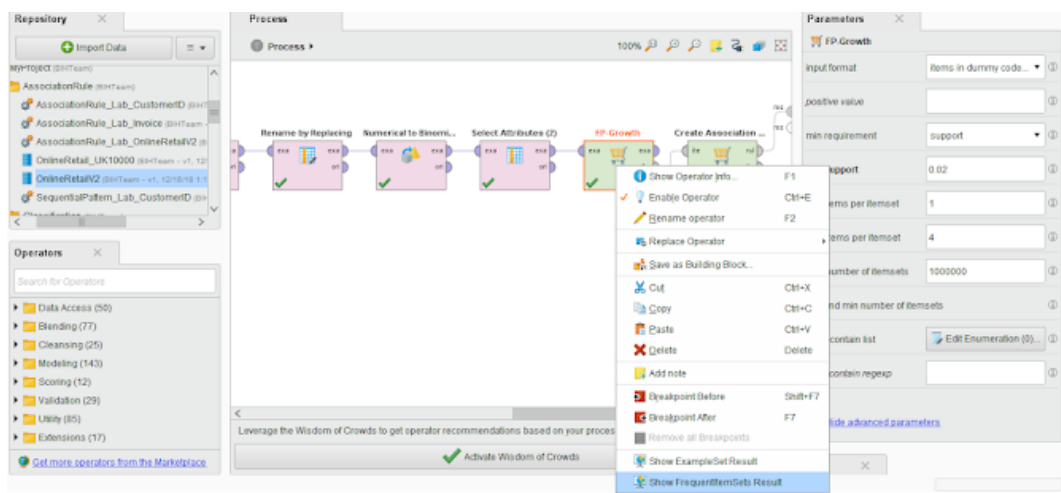
9.) การตั้งค่าในกล่อง FP-Growth



10.) การตั้งค่าในกล่อง Create Association Rules



11.) การดูผลทดสอบ FrequentItemSets ให้คลิกขวาที่กล่อง FP-Growth



คำอธิบายเพิ่มเติมประกอบ Lab: Association Rule Lab | Online Retail V2

1. ลักษณะ: Input

- เป็นลักษณะการ Transaction ของรายการซื้อของต่างๆ หรือ อาจจะเป็นรายการ การซื้อสินค้า ในใบเสร็จ โดย ข้อมูลจะถูกเก็บในแนวแถว โดย แสดงถึงการซื้อสินค้าแต่ละชิ้นที่ปรากฏในใบเสร็จนั้นๆ

2. Output ที่ได้

- จะได้เป็นกฎความสัมพันธ์สำหรับการซื้อสินค้า หากซื้อ Subset A แล้วจะซื้อ Subset B ตาม

3. Use Cases

- นำไปใช้ในการเสนอโปรโมชั่นการซื้อสินค้าต่างๆ ตามร้านค้า หรือ ห้างสรรพสินค้า หรือ ลองนำไปสร้างแพคเกจต่างๆ ให้กับลูกค้า

4. Limitations & Tips

- ภายในใบเสร็จ 1 ใบ (หรือใน 1 ธุรกรรม) ควรจะต้องมีสินค้าปรากฏมากกว่า 1 ชิ้นขึ้นไป