# Conducting Experiments to Build Models using Multiple Algorithms.

## Part 1: Clustering

Introduction to the dataset and its meta data.

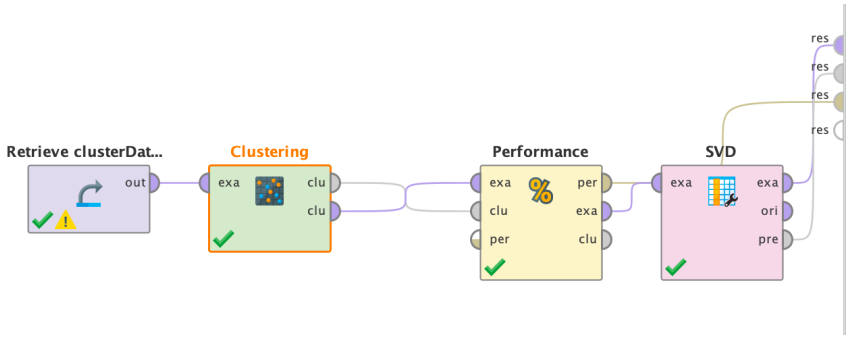
ไฟล์ที่นำมาใช้จัดกลุ่มข้อมูล ชื่อว่า “clusterDataset” เป็นนามสกุลไฟล์ .csv ข้อมูลที่ได้เป็นข้อมูลที่ใช้เพื่อทดสอบจากโปรแกรม RapidMiner Studio จาก ‘Generate Data’ Operator โดยใน Dataset มีทั้งหมด 4 Attributes และมีจำนวนแถวทั้งหมด 500 แถว

Clustering algorithms used

การสร้างโมเดลใช้ Algorithms 2 รูปแบบคือ k-Means กับ DBSCAN

Process (k-Means)

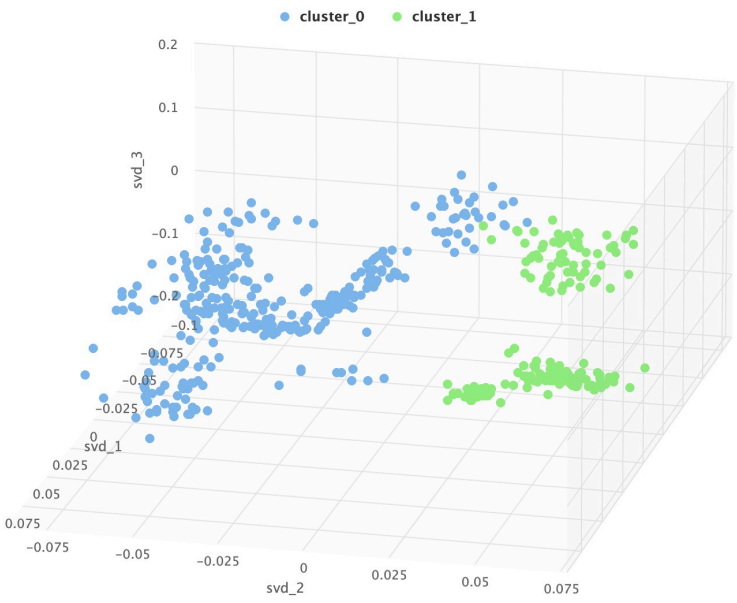
ประมวลผลโดย k-Means Operator วัดค่า Davies Bouldin จาก Cluster Distance Performance รวมถึงทำการลดขนาดมิติข้อมูลด้วย SVD Operator



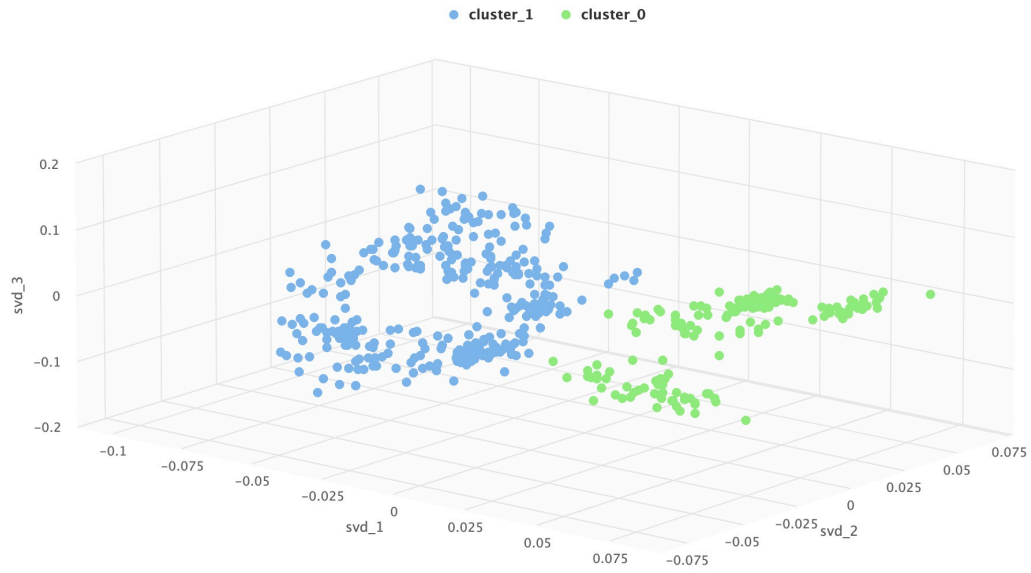

รูปภาพที่ 1 : แสดง Process ในโปรแกรม RapidMiner Studio

Subjective measures (k-Means)

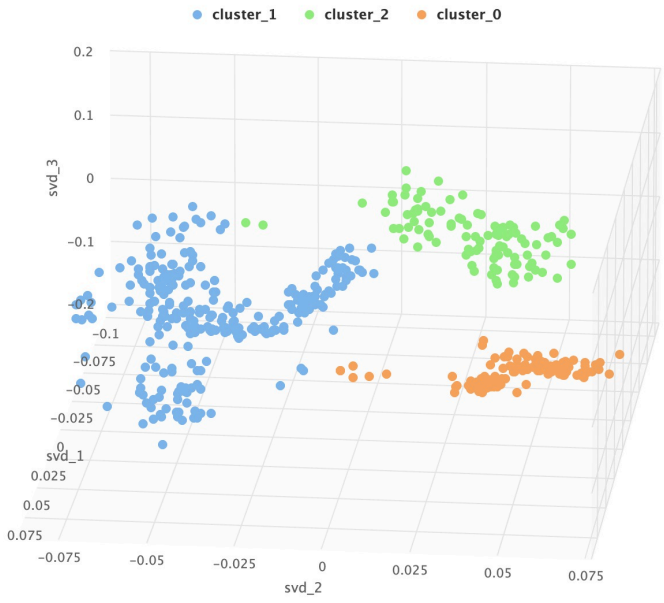
ผลลัพธ์ที่ได้จากการ Plot Graph แบบ Scatter Plot 3D โดยการเปลี่ยน Parameter ได้แก่ค่า k และรูปแบบวิธีการวัดระยะห่างแบบ Euclidian Distance และแบบ Manhattan Distance



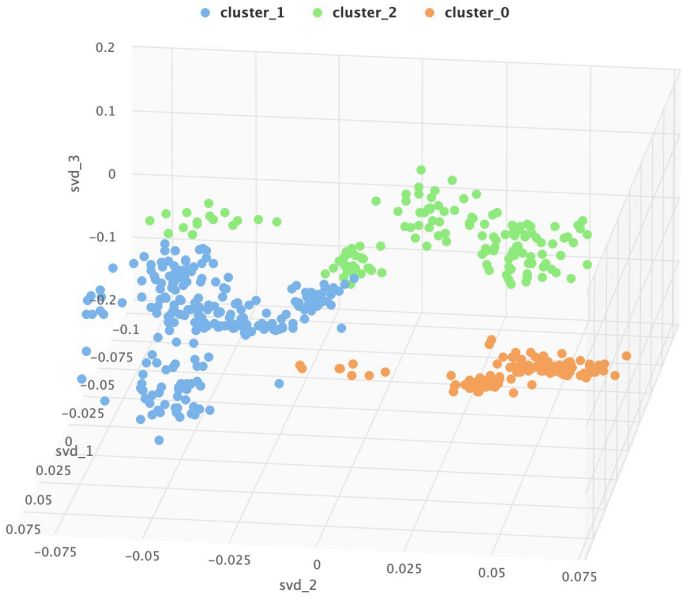

รูปภาพที่ 2 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า k = 2
โดยใช้วิธีการวัดระยะห่างแบบ Euclidian Distance



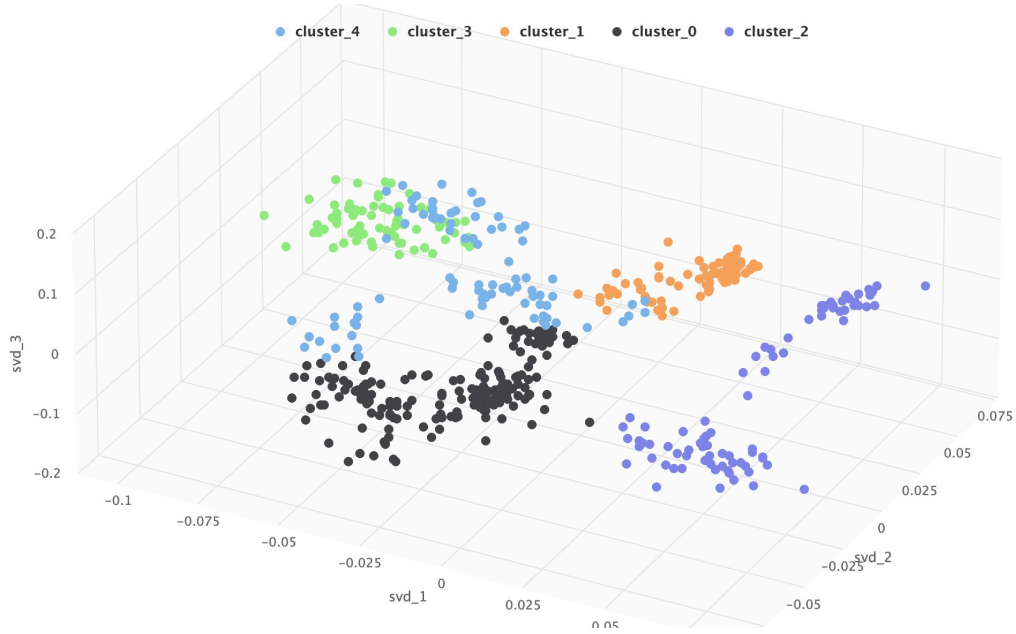

รูปภาพที่ 3 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า k = 2
โดยใช้วิธีการวัดระยะห่างแบบ Manhattan Distance



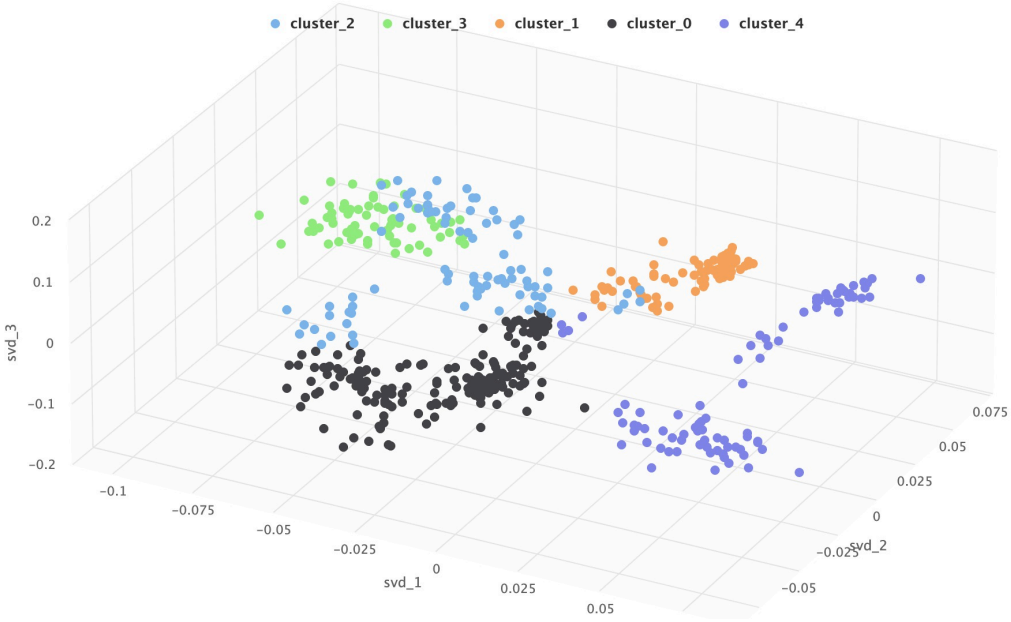

รูปภาพที่ 4 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า k = 3
โดยใช้วิธีการวัดระยะห่างแบบ Euclidian Distance




รูปภาพที่ 5 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า k = 3
โดยใช้วิธีการวัดระยะห่างแบบ Manhattan Distance




รูปภาพที่ 6 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า k = 5
โดยใช้วิธีการวัดระยะห่างแบบ Euclidian Distance




รูปภาพที่ 7 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า k=5
โดยใช้วิธีการวัดระยะห่างแบบ Manhattan Distance

ผลจากการวิเคราะห์แบบ Subjective Measure พบว่า ค่า k = 2 มีการจัดกลุ่มที่ชัดเจนมากที่สุด ดูจากระยะห่างระหว่างกลุ่มค่อนข้างแยกกันชัดเจน และวิธีการวัดระยะห่างทั้ง 2 รูปแบบมีผลลัพธ์ในการจัดกลุ่มที่ดีทั้งคู่

## Objective measures (k-Means)

ผลลัพธ์ที่ได้จากการประมวลผลโมเดล โดยใช้วิธีการคำนวณค่า Davies Bouldin และได้เปลี่ยน Parameter ค่า k เพื่อกำหนดค่า k และเลือกใช้วิธีการวัดระยะห่างแบบ Euclidian Distance

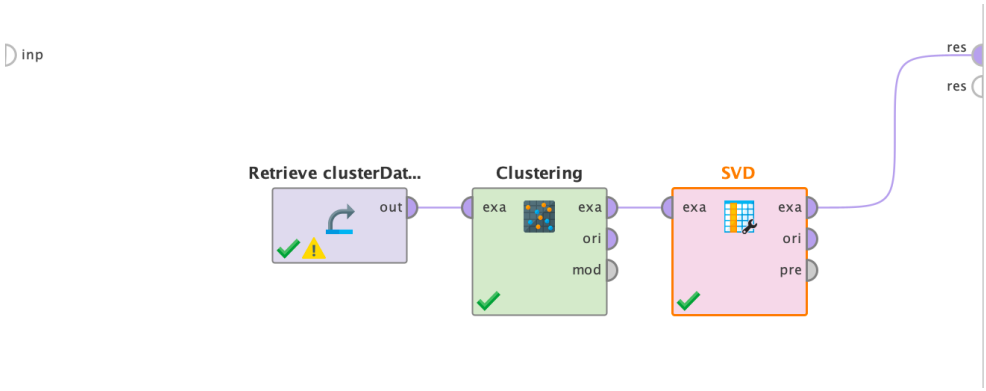
| ค่า k | ค่า Davies Bouldin |
|---|---|
| 2 | -1.519 |
| 3 | -1.102 |
| 4 | -0.987 |
| 5 | -0.952 |
| 6 | -0.908 |
| 8 | -0.819 |
| 10 | -0.703 |

รูปภาพที่ 8 : แสดงตารางเปรียบเทียบค่า Davies Bouldin กับค่า k ที่เปลี่ยนไป

ผลจากการวิเคราะห์แบบ Objective Measure พบว่า ค่า Davies Bouldin ยิ่งค่า k สูงขึ้น ค่า Davies Bouldin ยิ่งน้อยลง และแต่ละค่า k ค่อนข้างมีเอกลักษณ์การจัดกลุ่มที่ชัดเจน เมื่อใช้ Subjective measures ในการวิเคราะห์ร่วมด้วย

Process (DBSCAN)

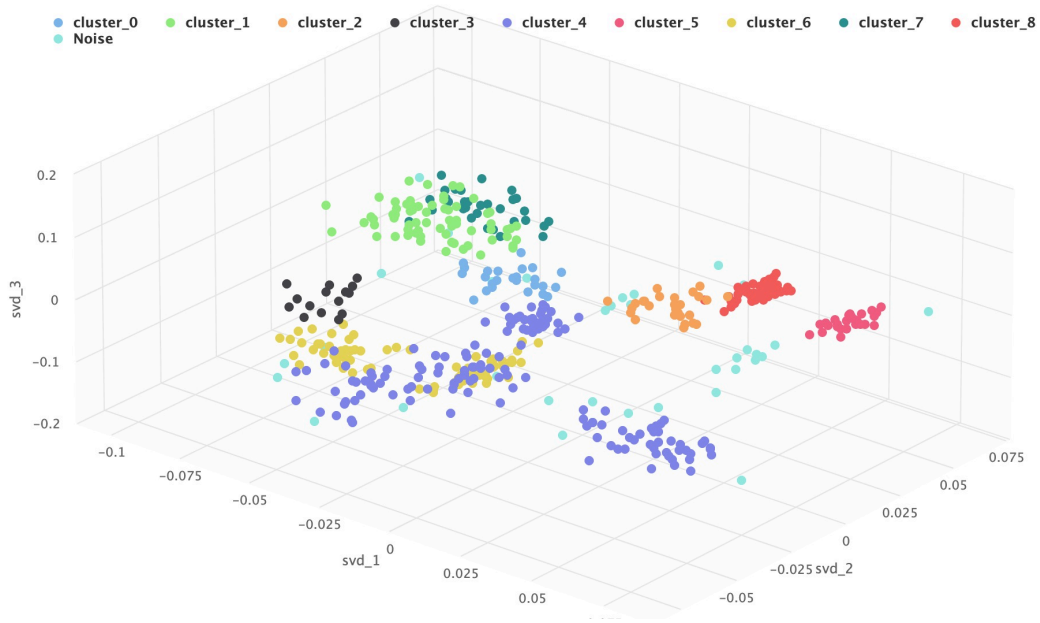
ประมวลผลโดย DBSCAN Operator ทำการลดขนาดมิติข้อมูลด้วย SVD Operator



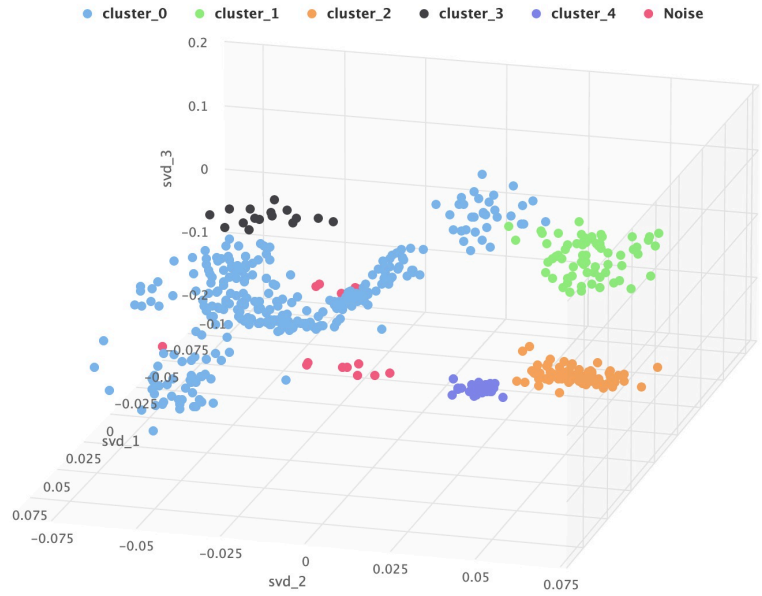

รูปภาพที่ 9 : แสดง Process ในโปรแกรม RapidMiner Studio

Subjective measures (DBSCAN)

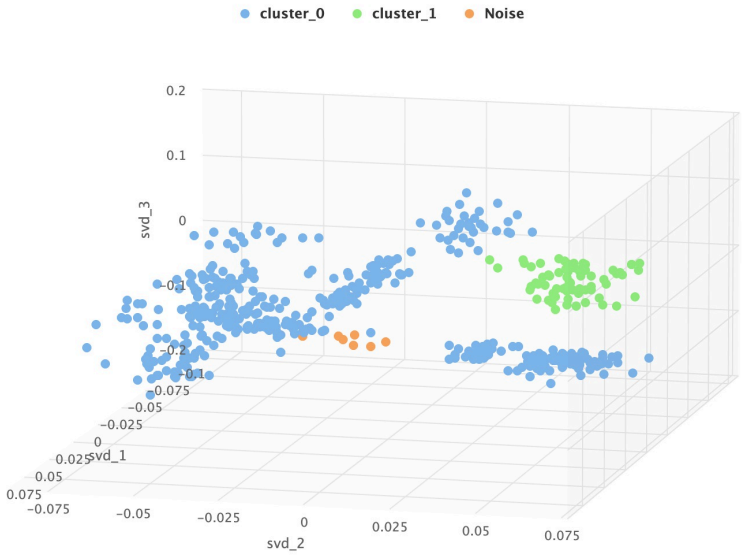
ผลลัพธ์ที่ได้จากการ Plot Graph แบบ Scatter Plot 3D โดยการเปลี่ยน Parameter ได้แก่ค่า epsilon และค่า min pts.



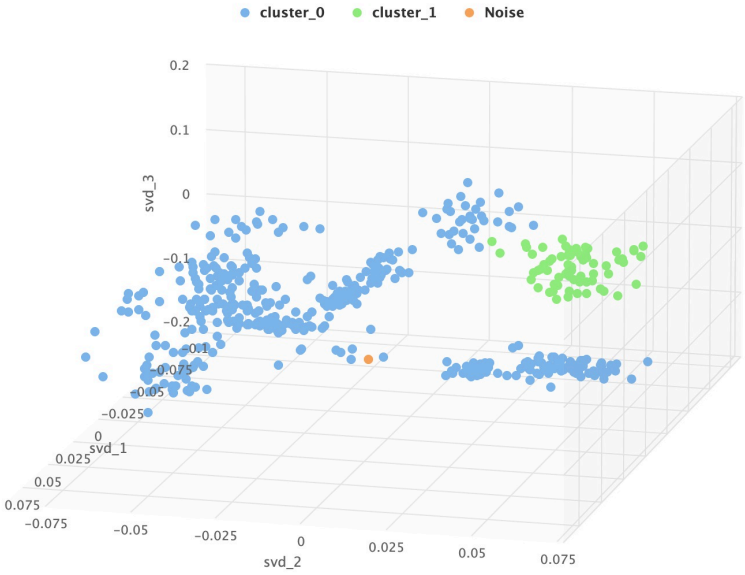

รูปภาพที่ 10 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า epsilon = 2 และค่า min pts. = 10



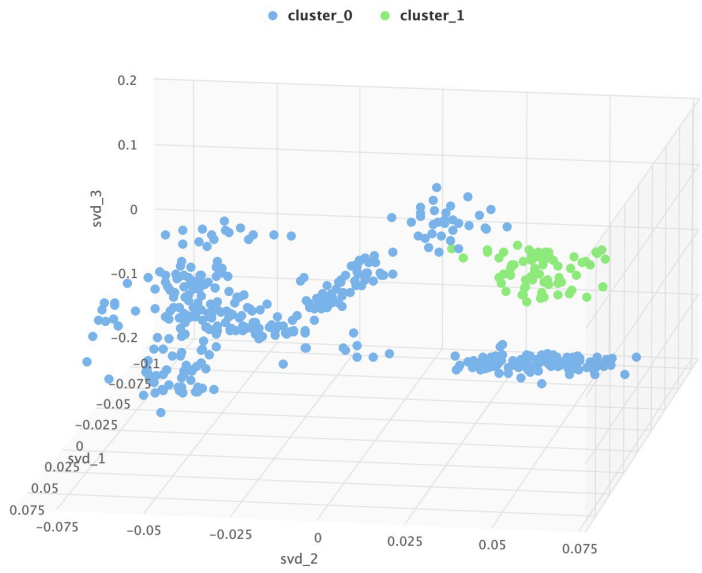

รูปภาพที่ 11 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า epsilon = 3 และค่า min pts. = 10



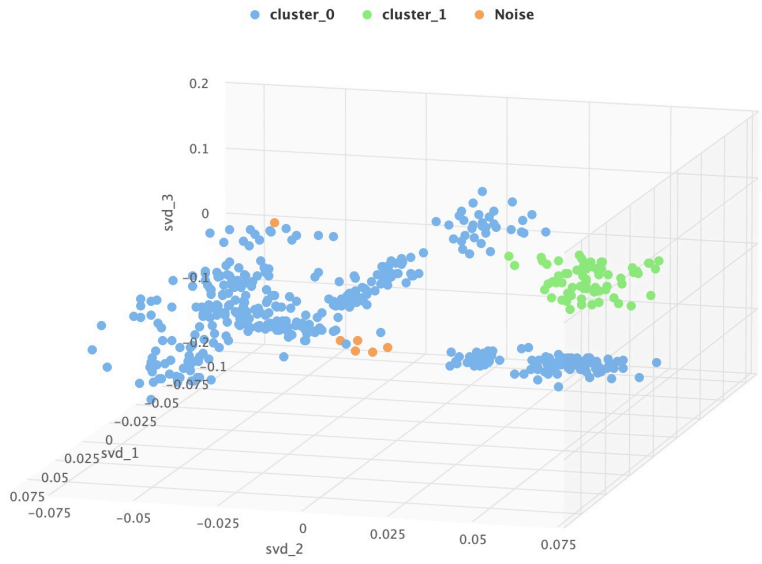

รูปภาพที่ 12 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า epsilon = 4 และค่า min pts. = 10




รูปภาพที่ 13 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า epsilon = 5 และค่า min pts. = 10




รูปภาพที่ 14 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า epsilon = 5 และค่า min pts. = 3




รูปภาพที่ 15 : แสดงกราฟแบบ Scatter Plot 3D ที่ค่า epsilon = 5 และค่า min pts. = 20

ผลจากการวิเคราะห์แบบ Subjective Measure พบว่า จำนวน Cluster = 2 มีการจัดกลุ่มที่ชัดเจนมากที่สุด ดูจากระยะห่างระหว่างกลุ่มค่อนข้างแยกกันชัดเจน โดยค่า Parameter ที่เลือกคือค่า epsilon = 5 และค่า min pts. = 3

สาเหตุของการเลือกค่า epsilon = 5 และค่า min pts. = 3 เพราะว่าการเลือก epsilon 5 ทำให้ได้จำนวน Cluster = 2 และมีระยะห่างระหว่างกลุ่มชัดเจน และสาเหตุที่เลือก min pts. = 3 เพราะว่า Noise ที่เกิดขึ้นใน min pts. อื่นๆ ระยะห่างระหว่างข้อมูลอื่นไม่ชัดเจน เลยเลือก min pts. = 3 เพื่อรวม Noise ให้อยู่ใน Cluster ด้วย

นอกจากนั้น พบว่าค่า Parameter ทั้งสอง ส่งผลต่อการเปลี่ยนแปลงของจำนวน Cluster และยังส่งผลต่อการ Detect Noise ด้วย ในทิศทางตรงกันข้ามกัน โดยหากยิ่งเพิ่มค่า epsilon มากขึ้น แสดงว่าจะมีโอกาสที่ Point จะจับคู่กันและเกิดเป็น Cluster ได้มากขึ้น (จากการเพิ่มระยะห่างที่ Core Point สามารถจับกับ Boarder Point ได้) แต่หากเพิ่มค่า Minimal Points มากขึ้น แสดงว่าจะมีโอกาสที่ Point จะจับคู่กันและเกิดเป็น Cluster ได้น้อยลง (จากการต้องเพิ่มจำนวน Boarder Point ที่ Core Point ต้องจับด้วย)

## Objective measures (DBSCAN)

ไม่สามารถทำในโปรแกรม RapidMiner Studio ได้ เนื่องจากการเปลี่ยน Output ของ DBSCAN Operator ทำให้ (clu) หรือ clustering หายไปและไม่สามารถเชื่อมกับ Cluster Density Performance Operator ได้

## Conclusion

จากการสร้างโมเดล Clustering โดยใช้ Algorithms k-Means และ DBSCAN โดยทำการทดลองเปลี่ยนค่า Parameter ต่างๆ เพื่อให้ได้ผลลัพธ์การจัดกลุ่มที่ชัดที่สุด พบว่าการจัดกลุ่มที่มีจำนวนกลุ่มเท่ากับ 2 ให้ผลลัพธ์การแยกกลุ่มออกจากกันได้ชัดเจนที่สุด

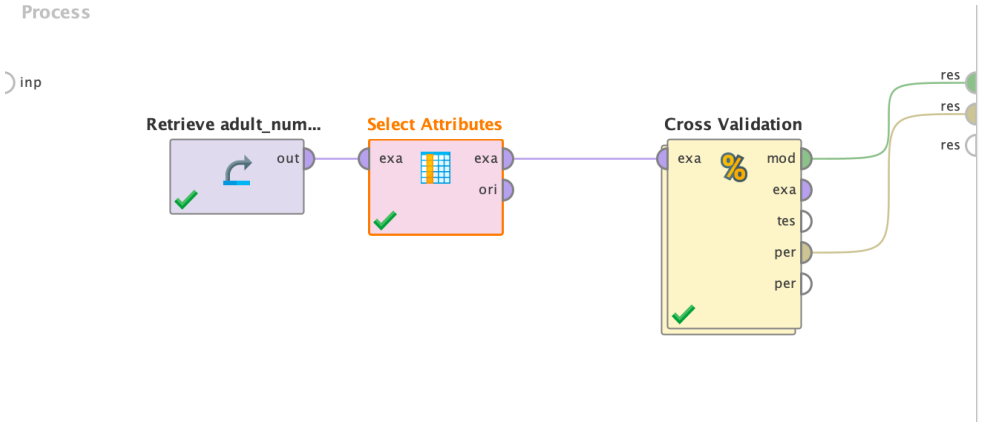
## Part 2: ANN

Introduction to the dataset and its meta data.

ไฟล์ที่นำมาใช้แบ่งกลุ่มข้อมูล ชื่อว่า “adult_numeric” เป็นนามสกุลไฟล์ .csv ข้อมูลผ่านการ Normalize มาแล้ว แบ่งกลุ่มข้อมูลตาม Label Class ตาม Income หรือรายได้ มี Attributes ทั้งหมด 6 Attributes มีข้อมูลทั้งหมด 1,000 แถว Attributes ได้แก่ age, fnlwgt, education_num, capital_gain, capital_loss, hours_per_week โดยมองว่ามี Attribute ที่ไม่เกี่ยวข้องอยู่คือ “fnlwgt” ต้องการนำออก เนื่องจากตัวแทนจำนวนประชากรไม่น่าเกี่ยวข้องโดยตรงกับรายได้

Process

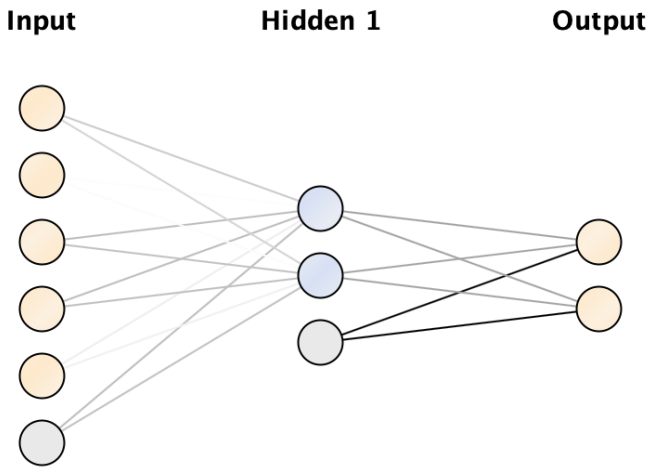
ประมวลผลโดย Neural Net Operator นำ Attribute ที่ไม่เกี่ยวข้องออกด้วย Select Attributes แบ่ง Training Data กับ Test Data ด้วย Cross Validation วัดค่า Accuracy ด้วย Performance Operator




รูปภาพที่ 16 : แสดง Process ในโปรแกรม RapidMiner Studio

Results of training using default parameters.

ผลลัพธ์ที่ได้จากการกำหนดค่า Parameter เป็น Default และกำหนด 1 Hidden Layer และ 2 Node




รูปภาพที่ 17 : แสดงกระบวนการ ANN โดยกำหนด 1 Hidden Layer 2 Node

**accuracy: 72.20% +/- 3.12% (micro average: 72.20%)**

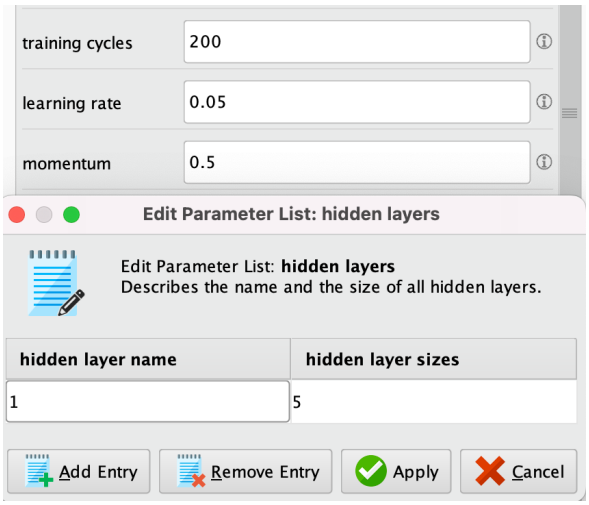
| | true >50K | true <=50K | class precision |
|---|---|---|---|
| pred. >50K | 363 | 142 | 71.88% |
| pred. <=50K | 136 | 359 | 72.53% |
| class recall | 72.75% | 71.66% | |

รูปภาพที่ 18 : แสดงผลลัพธ์ค่า Accuracy และตาราง Confusion Matrix

ผลการวิเคราะห์ พบว่าค่า Accuracy อยู่ในเกณฑ์ที่พอรับได้ แต่สามารถเพิ่มค่า Accuracy ได้มากกว่านี้จากการปรับเปลี่ยนค่า Parameter และเพิ่มจำนวน Hidden Layer กับจำนวน Node เพื่อเพิ่มความซับซ้อนให้กับโมเดล

Results when training using modified parameter settings.

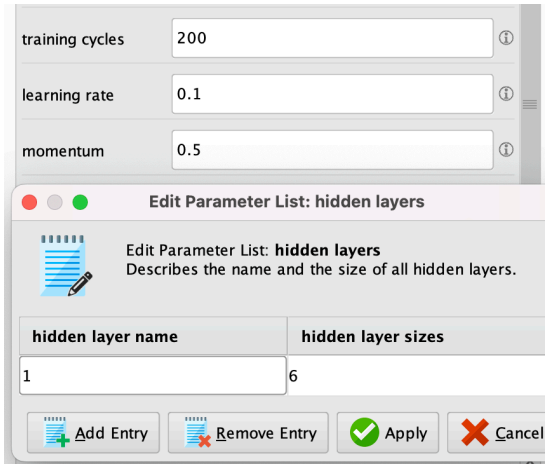
ผลลัพธ์จากการปรับแต่ง Parameter เพื่อเพิ่มค่า Accuracy ได้แก่ Learning Rate มีการปรับเพิ่มขึ้นเนื่องจากต้องการให้การ Adjust ค่า Weight ใหม่มีความเร็วมากขึ้น นอกจากนั้นมีการปรับค่า Momentum ให้ลดลง เพื่อลดอัตราการเร่งของ Learning Rate ให้ช้าลง และการเปลี่ยนเพิ่มจำนวน Hidden Layer และจำนวน Node ให้มากขึ้น เพื่อเพิ่มความซับซ้อนของโมเดล




**accuracy: 73.10% +/- 2.56% (micro average: 73.10%)**

| | true >50K | true <=50K | class precision |
|---|---|---|---|
| pred. >50K | 377 | 147 | 71.95% |
| pred. <=50K | 122 | 354 | 74.37% |
| class recall | 75.55% | 70.66% | |

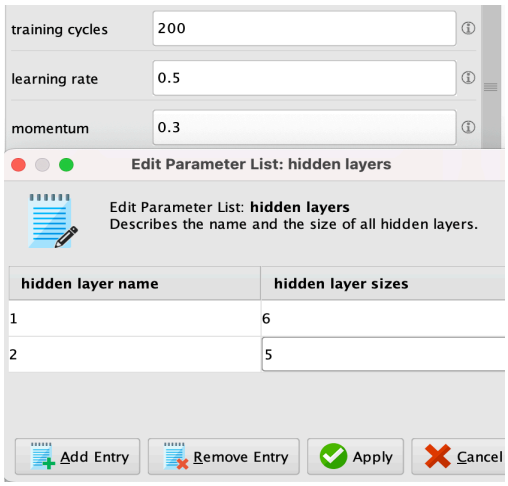
รูปภาพที่ 19 : แสดงการปรับเปลี่ยนค่า Parameter
และผลลัพธ์ค่า Accuracy และตาราง Confusion Matrix




**accuracy: 75.00% +/- 2.87% (micro average: 75.00%)**

| | true >50K | true <=50K | class precision |
|---|---|---|---|
| pred. >50K | 394 | true <=50K | 73.10% |
| pred. <=50K | 105 | 356 | 77.22% |
| class recall | 78.96% | 71.06% | |

รูปภาพที่ 20 : แสดงการปรับเปลี่ยนค่า Parameter
และผลลัพธ์ค่า Accuracy และตาราง Confusion Matrix




**accuracy: 71.30% +/- 3.20% (micro average: 71.30%)**

| | true >50K | true <=50K | class precision |
|---|---|---|---|
| pred. >50K | 365 | 153 | 70.46% |
| pred. <=50K | 134 | 348 | 72.20% |
| class recall | 73.15% | 69.46% | |

รูปภาพที่ 21 : แสดงการปรับเปลี่ยนค่า Parameter
และผลลัพธ์ค่า Accuracy และตาราง Confusion Matrix

ผลการวิเคราะห์ จากการเปลี่ยน Parameter ต่างๆนั้น ค่า Accuracy สูงสุด คือ 75% โดยการปรับ Parameter ได้แก่ Learning Rate = 0.1, Momentum = 0.05, Training Cycle = 200 และจำนวน 1 Hidden Layer กับ 6 Node

Training Cycle ไม่ได้มีการปรับเนื่องจากมองว่าจำนวนเท่านี้เหมาะสมแล้ว เพราะหากมากกว่านี้จะเกิด Overfitting ได้ และหากน้อยกว่านี้จะเกิด Underfitting ได้

การเพิ่มจำนวน Hidden Layer อาจจะไม่ตอบโจทย์ใน Dataset นี้ เนื่องจากมิติข้อมูลไม่ได้เยอะ และไม่จำเป็นต้องเพิ่มความซับซ้อนของโมเดล

## Conclusion

การเลือก Attributes ที่เกี่ยวข้องมีผลอย่างมากต่อผลลัพธ์ที่ได้ เนื่องจากหากใส่ Attributes ที่ไม่เกี่ยวข้อง ส่งผลทำให้ความแม่นยำในการประมวลผลนั้นต่ำลง

การปรับเปลี่ยน Parameter ต่างๆช่วยให้ค่า Accuracy นั้นสูงขึ้นจากเดิม และอยู่ในเกณฑ์ที่ดี และสามารถนำโมเดลไปใช้ทำนายผล Income ได้

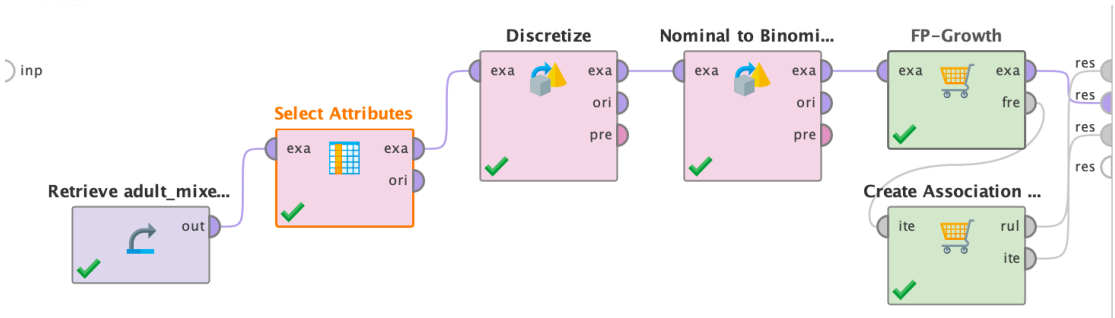
**Part3: Association Rules**

Introduction to the dataset and its meta data.

ไฟล์ที่นำมาใช้แบ่งกลุ่มข้อมูล ชื่อว่า “adult_mixed” เป็นนามสกุลไฟล์ .csv แบ่งกลุ่มข้อมูลตาม Label Class ตาม Income หรือรายได้ มี Attributes ทั้งหมด 14 Attributes มีข้อมูลทั้งหมด 1,000 แถว Attributes การเลือก Attribute ที่นำมาใช้ประมวล จากการวิเคราะห์มองว่ามี Attribute ที่ไม่เหมาะต่อการทำโมเดลได้แก่ fnlwgt, capital gain และ capital_loss ออก เนื่องจาก Values มีการกระจายตัวมากเกินไป และมีค่า 0 เยอะ

Process

ประมวลผลโดย FP-Growth Operator และสร้างกฎความสัมพันธ์ด้วย Create Association Rule Operator นอกจากนั้นได้นำ Attributes ที่ไม่เหมาะสมออกด้วย Select Attributes Operator ทำการแปลงข้อมูลที่เป็น Numeric ให้อยู่ในลักษณะของช่วงข้อมูล ด้วย Discretize Operator และแปลงข้อมูลทั้งหมดเป็น Binomial ด้วย Nominal to Binomial Operator (เนื่องจากแปลงแต่แรกไม่ได้เพราะโปรแกรมจะแปลงบาง Value ให้เป็น Missing Value เพื่อให้เหลือแค่ 2 ข้อมูลต่อ 1 Attribute)




รูปภาพที่ 22 : แสดง Process ในโปรแกรม RapidMiner Studio

Frequent Itemset Discussion

ผลจากการวิเคราะห์ พบว่ามี Frequent Itemset ที่มีความสำคัญต่อ Income อยู่ 8 Frequent Itemset โดยจะประกอบด้วย Itemset ที่น่าสนใจ ดังนี้ Native_Country ที่มี Value เป็น United=States มีค่า Support สูงสุด, Race ที่มี Value เป็น White, Workclass ที่มี Value เป็น Private และ Martiral_Status ที่มี Value เป็น Never-married ซึ่งหมายถึง Attributes ดังที่กล่าวมา มีความสำคัญกับ Income

| จำนวน Itemset | ค่า Support | Itemset ตัวที่ 1 | Itemset ตัวที่ 2 | Itemset ตัวที่ 3 | Itemset ตัวที่ 4 |
|---|---|---|---|---|---|
| 2 | 0.439 | native_country = United-States | label | | |
| 2 | 0.428 | race = White | label | | |
| 2 | 0.380 | workclass = Private | label | | |
| 2 | 0.220 | label | marital_status = Never-married | | |
| 3 | 0.385 | native_country = United-States | race = White | label | |
| 3 | 0.329 | native_country = United-States | workclass = Private | label | |
| 3 | 0.328 | race = White | workclass = Private | label | |
| 4 | 0.291 | native_country = United-States | race = White | workclass = Private | label |

รูปภาพที่ 23 : แสดงตาราง Frequent Itemset ร่วมกับค่า Support
โดยกำหนดให้ Label = “Income” Attribute

Rules Discussion

ผลจากการวิเคราะห์ เมื่อแสดงค่า Support, Confidence, Lift และ Conviction พบว่า Strong Association Rule มีเพียง 1 กฎคือ Martiral_Status ที่มี Value เป็น Never-married มีความสัมพันธ์มากๆกับ Income เนื่องจากค่า Confidence ใกล้เคียงกับ 1 รวมถึงค่า Lift กับ Conviction มีค่ามากกว่า 1 มาก

| Premises | Conclusion | Support | Confidence | Lift | Conviction |
|---|---|---|---|---|---|
| native_country = United-States, race = White | label | 0.385 | 0.476485149 | 0.951068161 | 0.953172577 |
| race = White | label | 0.428 | 0.486363636 | 0.970785701 | 0.971504425 |
| native_country = United-States | label | 0.439 | 0.495485327 | 0.988992669 | 0.989069351 |
| native_country = United-States, race = White, workclass = Private | label | 0.291 | 0.508741259 | 1.015451614 | 1.015758007 |
| race = White, workclass = Private | label | 0.329 | 0.522222222 | 1.042359725 | 1.044418605 |
| native_country = United-States, workclass = Private | label | 0.328 | 0.523125997 | 1.044163666 | 1.046397993 |
| workclass = Private | label | 0.38 | 0.530726257 | 1.059333846 | 1.063345238 |
| marital_status = Never-married | label | 0.22 | 0.873015873 | 1.742546653 | 3.929625 |

รูปภาพที่ 24 : แสดงตาราง Association Rule ร่วมกับค่าที่ใช้ประเมินต่างๆ
โดยกำหนดให้ Label = “Income” Attribute

## Part 4: Recommendation Engine

Introduction to the dataset and its meta data.

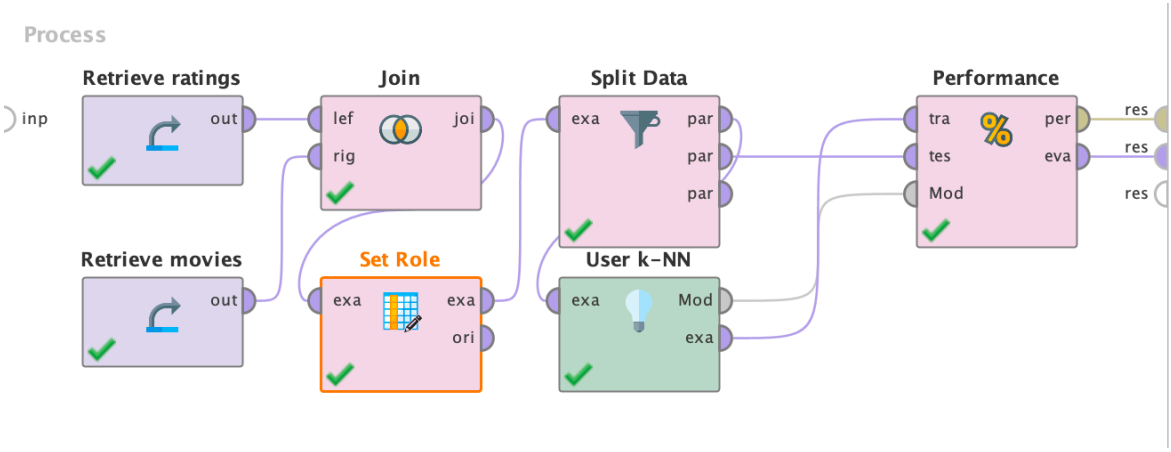
ไฟล์ที่นำมาใช้แบ่งกลุ่มข้อมูล ชื่อว่า “movies” กับ “ratings” เป็นนามสกุลไฟล์ .csv โดย “ratings” มี Attributes ทั้งหมด 4 Attributes 9,799 Row ส่วน “movies” มี Attributes ทั้งหมด 3 Attributes 9,742 Row มีการเลือก Attribute ที่นำมาใช้ประมวล จากการกำหนด UserID เป็น User Identification, MovieID เป็น Item Identification และ Rating เป็น Label

Recommendation algorithms used

การสร้างโมเดลใช้ Algorithm User-Based Nearest Neighbor

Process

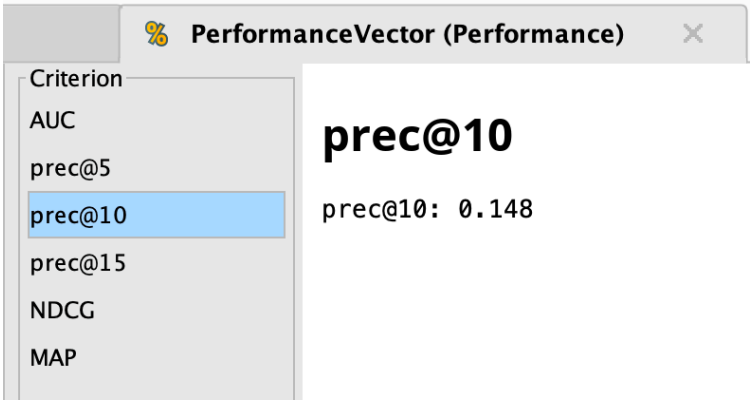
ประมวลผลโดย User k-NN Operator ใน Folder Collaborative Filtering Item Recommendation มี 2 Dataset ทำการใช้ Join Operator เพื่อผนวก Key Attribute เข้าด้วยกัน และใช้ Set Role Operator ในการเลือก User Identification กับ Item Identification พร้อมใช้ Split Data เพื่อแยก Training Data กับ Test Data สุดท้ายใช้ Performance Item Recommendation ในการหาค่า Prec@10




รูปภาพที่ 25 : แสดง Process ในโปรแกรม RapidMiner Studio

Results of training using default parameters.

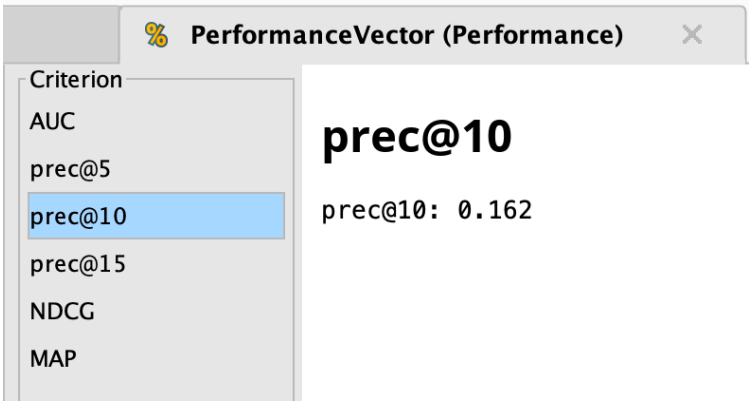
ผลจากการใช้ Parameter ที่เป็นค่า Default พบว่ามีค่าความแม่นยำต่ำในการ Recommendation หนัง 10 เรื่อง คิดเป็น Percent เพียง 14.8%



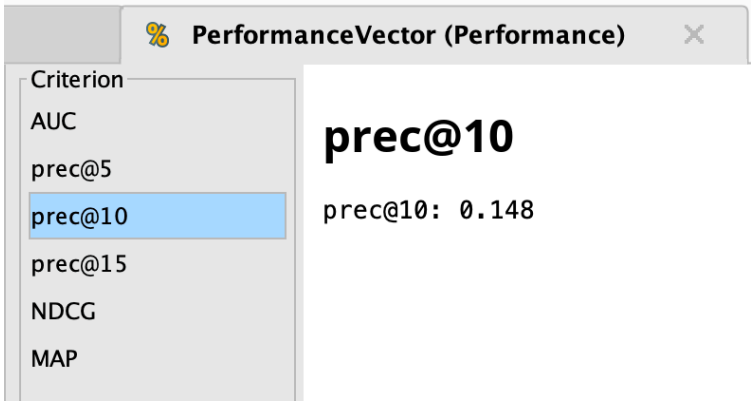

รูปภาพที่ 26 : แสดงค่า prec@10 จากการตั้งค่า Parameter แบบ Default

Results when training using modified parameter settings.

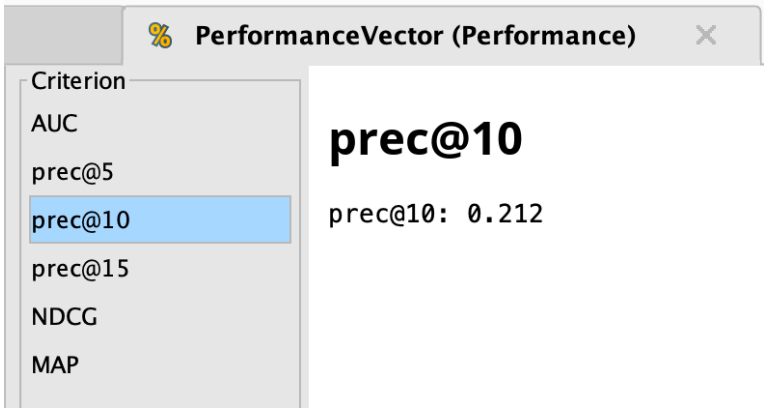
ผลลัพธ์จะได้ค่า prec@10 ที่เปลี่ยนแปลงไป



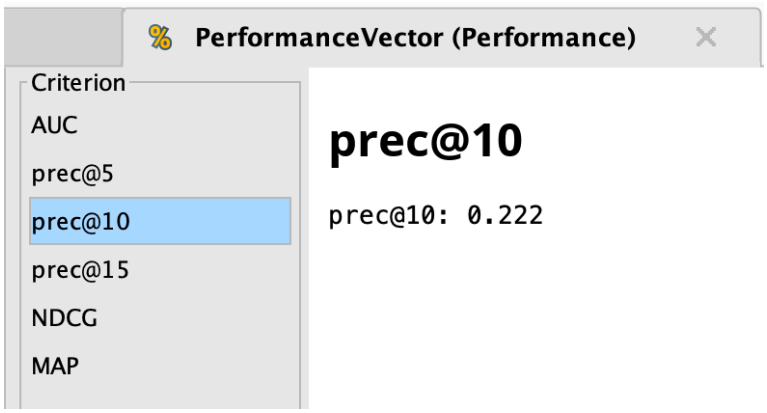

รูปภาพที่ 27 : แสดงค่า prec@10 จากการตั้งค่า k=50




รูปภาพที่ 28 : แสดงค่า prec@10 จากการตั้งค่า k=90




รูปภาพที่ 29 : แสดงค่า prec@10 จากการตั้งค่า k=30




รูปภาพที่ 30 : แสดงค่า prec@10 จากการตั้งค่า k=10

ผลจากการวิเคราะห์พบว่าการเปลี่ยน Parameter ค่า k=10 มีค่าความแม่นยำในการ Recommendation หนัง 10 เรื่อง ให้กับ User มากที่สุดที่ 22.2%

Results 10 Movie Recommendation

ผลลัพธ์จากการใช้ Apply Model Operator เพื่อจัดอันดับ 10 หนังที่จะ Recommend ได้ผลลัพธ์ดังนี้

| user_id | item_id | rank ↑ |
|---|---|---|
| 1 | 924 | 1 |
| 2 | 4993 | 1 |
| 3 | 858 | 1 |
| 4 | 1136 | 1 |
| 5 | 356 | 1 |
| 6 | 318 | 1 |
| 7 | 260 | 1 |
| 8 | 595 | 1 |
| 9 | 260 | 1 |
| 10 | 79132 | 1 |

| | |
|---|---|
| 924 | 2001: A Space Odyssey (1968) |
| 4993 | Lord of the Rings: The Fellowship of the Ring, |
| 858 | Godfather, The (1972) |
| 1136 | Monty Python and the Holy Grail (1975) |
| 356 | Forrest Gump (1994) |
| 318 | Shawshank Redemption, The (1994) |
| 260 | Star Wars: Episode IV - A New Hope (1977) |
| 595 | Beauty and the Beast (1991) |
| 260 | Star Wars: Episode IV - A New Hope (1977) |
| 79132 | Inception (2010) |

รูปภาพที่ 31 : แสดงอันดับของหนัง และชื่อหนังในแต่ละอันดับ

Conclusion

จากการใช้ Dataset เพื่อสร้างโมเดล Recommend หนังให้กับ User นั้น โดยใช้ Algorithm User-Based Nearest Neighbor นั้น ผลลัพธ์ที่ได้มีความแม่นยำเพียง 22.2% ในการแนะนำหนังทั้งหมด 10 เรื่องให้กับ User