บทที่ 2 ข้อมูล

หัวข้อ

2.1 ประเภทของข้อมูล

2.2 คุณภาพข้อมูล

2.3 การเตรียมข้อมูลก่อนการประมวลผล

2.4 การวัดความคล้ายคลึงและความแตกต่าง

2.1 ประเภทของข้อมูล (Types of Data)

2.1.1 แอทริบิวต์และการวัด

คำนิยาม 2.1. ชุดข้อมูล (data set) คือกลุ่มของดาต้าอ็อบเจ็กต์ **(data objects)** ซึ่งประกอบด้วยแอทริบิวต์ต่าง ๆ **(attributes)** ที่อธิบายคุณลักษณะของดาต้าอ็อบเจ็กต์

ตัวอย่างที่ 2.1 (ชุดข้อมูลนิสิต) ชุดข้อมูลมักถูกจัดเก็บไว้ในไฟล์ซึ่งประกอบด้วยเรคอร์ด (หรือแถว) แต่ละเรคอร์ดแทนข้อมูลเกี่ยวกับอ็อบเจ็กต์แต่ละตัวที่เราสนใจ เช่น ชุดข้อมูลนิสิตในตารางที่ 2.1 ประกอบด้วยเรคอร์ดจำนวน 5 เรคอร์ด แต่ละเรคอร์ดประกอบด้วยแอทริบิวต์ (หรือคอลัมน์) ที่อธิบายคุณลักษณะของนิสิต (หรือ **data object)** แต่ละคน จำนวน **4** แอทริบิวต์ (หรือคอลัมน์) ดังนี้คือ ลำดับที่ รหัสนิสิต ชั้นปี และเกรดเฉลี่ยสะสม เช่น เรคอร์ดที่ 2 เป็นข้อมูลเกี่ยวกับนิสิต **(**หรือ **data object)** รหัส 1034262 ชั้นปีที่ 3 มีเกรดเฉลี่ยสะสมเท่ากับ 3.24 เป็นต้น

ตารางที่ 2.1. ชุดข้อมูลนิสิต (Student Information Data Set)

แอทริบิวต์ (Attributes)

ดาต้าอ็อบเจ็กต์ (Data objects)

ลำดับที่	รหัสนิสิต	ชั้นปี	เกรดเฉลี่ยสะสม
1	1034261	2	2.75
2	1034262	3	3.24
3	1034263	2	3.51
4	1034265	1	2.99
5	1034266	3	3.12

คำนิยาม 2.2. แอทริบิวต์ (Attributes) คือคุณสมบัติหรือคุณลักษณะของดาต้าอ็อบเจ็กต์แต่ละตัว แอทริบิวต์เดียวกันของดาต้าอ็อบเจ็กต์คนละตัวอาจมีค่าแตกต่างกัน และค่าแอทริบิวต์ของดาต้าอ็อบเจ็กต์ก็อาจมีค่าเปลี่ยนไปได้ตามเวลา เช่น แอทริบิวต์อุณหภูมิ ณ ผิวน้ำทะเล ในพื้นที่ต่าง ๆ จะมีค่าเป็นตัวเลขซึ่งแตกต่างกันไปในแต่ละพื้นที่ และแต่ละช่วงเวลา เป็นต้น

คำนิยาม 2.3. สเกลการวัด (measurement scales) คือการกำหนดตัวเลขหรือสัญลักษณ์อย่างมีระบบให้กับแอทริบิวต์ของดาต้าอ็อบเจ็กต์

ประเภทของแอทริบิวต์ (หรือประเภทของสเกลการวัด)

แอทริบิวต์หนึ่งตัวสามารถอธิบายได้ด้วยสเกลการวัดหลายสเกล และค่าหรือสัญลักษณ์ของสเกลที่ถูกกำหนดให้กับแอทริบิวต์อาจจะมีคุณสมบัติตรงกันกับแอทริบิวต์หรือไม่ก็ได้

ตัวอย่างที่ 2.2 สเกลการวัดจำนวนเต็ม (integer) ถูกใช้ในการกำหนดค่าของแอทริบิวต์รหัสพนักงาน (Employee ID) และอายุของพนักงาน (Employee Age)

เมื่อเราใช้จำนวนเต็มแทนค่าของแอทริบิวต์ Employee ID โอเปอเรชั่นที่ใช้ดำเนินการกับค่าจำนวนเต็มของแอทริบิวต์นี้ได้ มีเพียงโอเปอเรชั่นเดียวคือ การทดสอบว่ารหัสพนักงานสองรหัสมีค่าเท่ากันหรือไม่ (equality test)

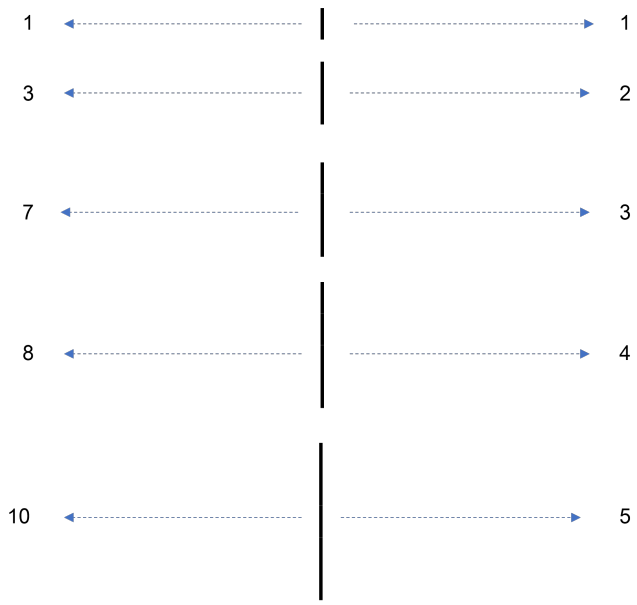
แต่เมื่อเราใช้จำนวนเต็มแทนค่าของแอทริบิวต์ Employee Age โอเปอเรชั่นที่ใช้ดำเนินการกับค่าจำนวนเต็มของแอทริบิวต์นี้ได้ จะมีหลายโอเปอเรชั่น เช่น การหาค่าเฉลี่ย, ค่าต่ำสุด, ค่าสูงสุด และการทดสอบค่า เป็นต้น

ตัวอย่างที่ 2.3 แสดงตัวอย่างการใช้สเกลการวัดแบบจำนวนเต็ม 2 ชนิด ในการกำหนดค่าให้กับเส้นตรง

สเกลทางด้านซ้ายมือกำหนดค่าจำนวนเต็มให้กับเส้นตรงโดยดูจากอันดับของความยาวของเส้นตรงเรียงจากน้อยไปมาก เส้นตรงที่มีความยาวมากกว่า จะมีค่าแอทริบิวต์ที่มากกว่า

สเกลทางด้านขวามือกำหนดค่าจำนวนเต็มให้กับเส้นตรงโดยใช้จำนวนเท่าของความยาวของเส้นตรงเมื่อเทียบกับความยาวของเส้นตรงเส้นแรกด้านบนสุดของรูป

จะเห็นได้ว่าสเกลทางด้านซ้ายมือ สื่อถึงเฉพาะคุณสมบัติการเรียงลำดับ (ordering property) ของความยาวของเส้นตรง ในขณะที่สเกลทางด้านขวามือสามารถบันทึกได้ทั้งคุณสมบัติการเรียงลำดับ (ordering property) และคุณสมบัติการบวก (additivity property) ของความยาวของเส้นตรง



รูปที่ 1 การวัดความยาวของส่วนของเส้นตรงด้วยสเกลที่ต่างกัน 2 ชนิด

การแบ่งประเภทของแอทริบิวต์ด้วยโอเปอเรชั่นของระบบจำนวน

โอเปอเรชั่นของระบบจำนวนที่มักถูกนำมาใช้ในการจำแนกประเภทแอทริบิวต์ ได้แก่ (1) Distinctness (= และ ≠) (2) Order (<, ≤ , >, ≥) (3) Addition (+ และ -) (4) Multiplication (x และ /) โดยเราสามารถแบ่งประเภทของแอทริบิวต์ออกได้เป็น 4 ประเภท คือ Nomial, Ordinal, Interval, Ratio ดังแสดงในตารางที่ 2.2

ตารางที่ 2.2. ชนิดของแอทริบิวต์

ชนิดของแอทริบิวต์		คำอธิบาย	ตัวอย่าง	โอเปอเรชั่น
Categorical (เชิงคุณภาพ)	**Nominal**	ค่าของ Nominal attribute สามารถใช้ในการแยกแยะดาต้าอ็อบเจ็กต์ได้ ด้วยโอเปอเรชั่น Distinctness (=, ≠)	รหัสไปรษณีย์ รหัสพนักงาน สีตา เพศ	ฐานนิยม, entropy, contingency correlation, Chi-squared test
	Ordinal	มีคุณสมบัติและโอเปอเรชั่น Distinctness เช่นเดียวกันกับ Nominal attributes และค่าของ Ordinal attribute สามารถใช้ในการเรียงลำดับดาต้าอ็อบเจ็กต์ได้ ด้วยโอเปอเรชั่น Order (<, >)	ความแข็งของแร่ธาตุ, เกรด {A, B+, B, C+, C, D+, D, F},	มัธยฐาน, เปอร์เซ็นต์ไทล์, rank correlation, run tests, sign tests
Numeric (เชิงปริมาณ)	**Interval**	มีคุณสมบัติและโอเปอเรชั่น Distinctness และ Order เช่นเดียวกันกับ Nominal attributes และ Ordinal attributes นอกจากนี้ ความแตกต่างระหว่าง interval attributes สองค่า คำนวณได้ด้วยโอเปอเรชั่น Addition (+, -) สามารถตีความได้ กล่าวคือ interval attributes จะมีหน่วยของการวัด	อุณหภูมิในหน่วยองศาเซลเซียส หรือองศาฟาเรนไฮต์, วันที่ตามปฏิทิน	ค่าเฉลี่ย, ส่วนเบี่ยงเบนมาตรฐาน, Pearson's correlation, t-test, F-test
	Ratio	มีคุณสมบัติและโอเปอเรชั่น Distinctness, Order, และ Interval เช่นเดียวกันกับ Nominal attributes, Ordinal attributes, และ Interval attributes นอกจากนี้ อัตราส่วนของ ratio attributes ซึ่งคำนวณได้โดยใช้โอเปอเรชั่น Multiplication (x, /) สามารถตีความได้	อุณหภูมิในหน่วยเคลวิน (Kelvin), อายุ, มวล, ความยาว, กระแสไฟฟ้า	ค่าเฉลี่ยเรขาคณิต, ค่าเฉลี่ยฮาร์มอนิค, เปอร์เซ็นต์ความผันแปร

การแบ่งประเภทของแอทริบิวต์ด้วยจำนวนของค่าที่เป็นไปได้

Discrete Attributes. คือแอทริทบิวต์ที่มีจำนวนค่าที่เป็นไปได้จำกัดหรือไม่จำกัดแต่สามารถนับแจกแจงได้ (finite or countably infinite) ส่วนใหญ่มักเป็นแอทริบิวต์เชิงคุณภาพ (nominal, ordinal) เช่น รหัสไปรษณีย์ รหัสพนักงาน เป็นต้น แอทริบิวต์ชนิดนี้มักถูกแทนค่าด้วยตัวแปรที่มีชนิดเป็นจำนวนเต็ม (integer variables) **Binary attributes** คือดีสครีตแอทริบิวต์ชนิดพิเศษที่มีค่าที่เป็นไปได้เพียงสองค่าเท่านั้น เช่น จริง/เท็จ, ใช่/ไม่ใช่, ชาย/หญิง, หรือ 0/1 เป็นต้น

Continuous Attributes. คือแอทริบิวต์ที่มีค่าเป็นจำนวนจริง ส่วนใหญ่มักเป็นแอทริบิวต์เชิงปริมาณ (interval, ratio) เช่น อุณหภูมิ, ความสูง, น้ำหนัก แอทริบิวต์ชนิดนี้มักถูกแทนค่าด้วยตัวแปรที่มีชนิดจุดลอยตัว (floating-point variables)

แอทริบิวต์แบบไม่สมมาตร (Asymmetric Attributes)

คือ แอทริบิวต์ที่การปรากฏของค่าเท่านั้นที่มีความสำคัญ เช่น ดาต้าเซตที่บันทึกการลงทะเบียนเรียนของนิสิตในแต่ละวิชาที่เปิดในแต่ละภาคการศึกษา ในกรณีนี้แอทริบิวต์แต่ละรายวิชาของข้อมูลของนิสิตแต่ละคนจะมีค่าเป็น 1 ก็ต่อเมื่อนิสิตคนนั้นได้ลงทะเบียนเรียนในรายวิชานั้น เนื่องจากนิสิตแต่ละคนจะลงทะเบียนเพียงไม่กี่รายวิชา จากวิชาที่เปิดสอนทั้งหมด ค่าของแอทริบิวต์ส่วนใหญ่จะมีค่าเป็น 0 ดังนั้นการประมวลผลดาต้าเซตนี้ให้ได้อย่างมีประสิทธิภาพจึงควรมุ่งเน้นที่ค่าที่ไม่เป็นศูนย์กล่าวคือ เฉพาะรายวิชาที่นิสิตได้มีการลงทะเบียนเรียนนั่นเอง แอทริบิวต์แบบไม่สมมาตรที่มีค่าที่เป็นไปได้สองค่า จะเรียกว่า **asymmetric binary attributes** แอทริบิวต์ชนิดนี้จะมีความสำคัญมากในการวิเคราะห์ความสัมพันธ์ (association analysis) ซึ่งเราจะได้ศึกษาในบทที่ 4 ต่อไป

2.1.2 ชนิดของดาต้าเซต

ชนิดของดาต้าเซตสามารถแบ่งได้เป็น 3 ประเภทหลัก คือ ข้อมูลเรคอร์ด (record data) ข้อมูลกราฟ (graph based data) และข้อมูลที่มีลำดับ (ordered data) ก่อนที่จะอธิบายคุณสมบัติของดาต้าเซตแต่ละชนิด เราจะพิจารณาคุณลักษณะสามประการของดาต้าเซตที่มีผลกระทบอย่างมีนัยสำคัญต่อเทคนิคการทำเหมืองข้อมูลที่นำมาใช้ ดังนี้คือ มิติของข้อมูล (dimensionality), การกระจายตัว (distribution), และระดับความละเอียดของข้อมูล (resolution)

- **Dimensionality** มิติของดาต้าเซตคือ จำนวนของแอทริบิวต์ของอ็อบเจ็กต์ในดาต้าเซต การวิเคราะห์ข้อมูลที่มีมิติสูง (high-dimensional data) มีความยากและท้าทายมากและมีชื่อเรียกปัญหาการวิเคราะห์ที่เกิดจากข้อมูลที่มีมิติสูงว่า **the curse of dimensionality** เพื่อลดปัญหาดังกล่าว ในการทำเหมืองข้อมูลเราจึงมักทำการประมวลผลข้อมูลในเบื้องต้นโดยใช้เทคนิคการลดจำนวนมิติของข้อมูล (**dimensionality reduction**) ต่าง ๆ เช่น principal component analysis (PCA) เป็นต้น
- **Distribution** การกระจายของดาต้าเซต คือ ความถี่ของการเกิดขึ้นของค่าต่าง ๆ ของแอทริบิวต์ของอ็อบเจ็กต์ในดาต้าเซตนั้น แม้ว่าในวิชาสถิติจะมีการกระจายข้อมูลแบบมาตรฐาน หลายแบบ เช่น Gaussian distribution, Poisson distribution แต่ดาต้าเซตที่พบในการทำเหมืองข้อมูลมักไม่สามารถอธิบายได้โดยใช้การกระจายเชิงสถิติแบบมาตรฐาน ดังนั้นอัลกอริทึมการทำเหมืองข้อมูลส่วนใหญ่จึงมักจะไม่ตั้งสมมติฐานเกี่ยวกับการกระจายของดาต้าเซตเอาไว้ อย่างไรก็ตามคุณสมบัติที่เกี่ยวกับการกระจายตัวบางประการเช่น ความเบ้ (skewness) และ ความกระจัดกระจาย (sparsity) ก็มักจะมีผลกระทบอย่างมากต่ออัลกอริทึมการทำเหมืองข้อมูล
- **Resolution** รูปแบบของข้อมูลขึ้นอยู่กับระดับความละเอียดของข้อมูลที่จัดเก็บ เช่น อุณหภูมิ ณ ระดับผิวน้ำทะเล ในแต่ละจุดที่ความละเอียดในระดับ 1 เมตร จะค่อนข้างราบเรียบไม่แตกต่างกันมากนัก แต่หากลดความละเอียดของข้อมูลลงไปเป็นที่ทุก ๆ 10 กิโลเมตร จะพบว่าค่าอุณหภูมิมักจะมีความแตกต่างกันอย่างเห็นได้ชัด

ข้อมูลเรคอร์ด (Record Data)

ข้อมูลที่ใช้ในการทำเหมืองข้อมูลมักจะอยู่ในรูปแบบของกลุ่มของเรคอร์ดข้อมูล (data objects) ซึ่งแต่ละเรคอร์ดประกอบด้วยแอทริบิวต์หรือฟิลด์ข้อมูลที่คงที่ ดังตัวอย่างในรูปที่ 2(a) แสดงตัวอย่างข้อมูลเรคอร์ดแบบพื้นฐาน แอทริบิวต์แต่ละตัวและเรคอร์ดแต่ละเรคอร์ดในดาต้าเซตจะไม่มีความสัมพันธ์ต่อกันอย่างเด่นชัด และเรคอร์ดแต่ละเรคอร์ดจะประกอบด้วยแอทริบิวต์เซตเดียวกัน ข้อมูลเรคอร์ดมักจะเก็บอยู่ในรูปแบบของ flat files หรือในฐานข้อมูลเชิงสัมพันธ์ (relational database)

Tid	Refund	Marital Status	Taxable Income	Defaulted Borrower
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

(a) Record data.

TID	*ITEMS*
1	Bread, Soda, Milk
2	Beer, Bread
3	Beer, Soda, Diapers, Milk
4	Beer, Bread, Diapers, Milk
5	Soda, Diapers, Milk

(b) Transaction data.

Projection of x Load	Projection of y Load	Distance	Load	Thickness
10.23	5.27	15.22	27	1.2
12.65	6.25	16.22	22	1.1
13.54	7.23	17.34	23	1.2
14.27	8.43	18.45	25	0.9

(c) Data matrix.

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

(d) Document-term matrix.

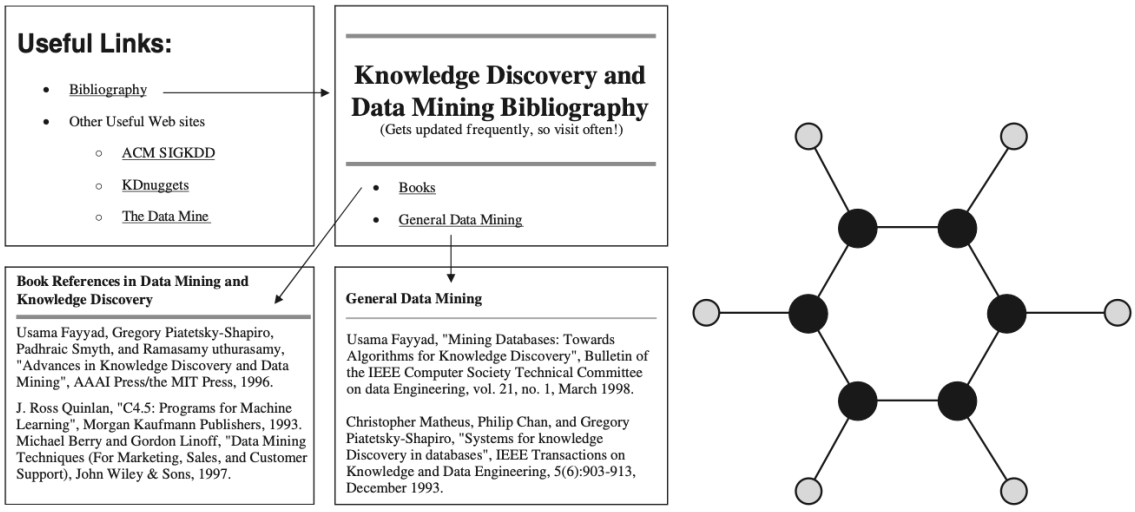
รูปที่ 2 ข้อมูลเรคอร์ดชนิดต่าง ๆ

- **Transaction or Market Basket Data** ข้อมูลทรานแซกชันคือข้อมูลเรคอร์ดชนิดพิเศษ ซึ่งแต่ละเรคอร์ด (ทรานแซกชัน) คือเซตของ data item ที่มีความสัมพันธ์กัน เช่น ข้อมูลทรานแซกชันหนึ่งของ ณ จุดขาย (Point-of-Sales) ของร้านสะดวกซื้อจะประกอบด้วย data itemของรายการสินค้าที่ลูกค้าเลือกซื้อและนำมาจ่ายเงิน เช่น โซดา, ผ้าอ้อม, นม เป็นต้น รูปที่ 2(b) แสดงตัวอย่างข้อมูลเรคอร์แบบทรานแซกชันของร้านสะดวกซื้อแห่งหนึ่ง ข้อมูลทรานแซกชัน เรียกอีกอย่างว่าข้อมูลตะกร้าสินค้า (market basket data) คือกลุ่มของเซตของ data items แต่ละเซตเปรียบได้กับเรคอร์ดแต่ละเรคอร์ด ซึ่งประกอบด้วยฟิลด์ของแอทริบิวต์แบบไม่สมมาตร

- **The Data Matrix** ถ้าดาต้าอ็อบเจ็กต์ทุกตัวในชุดข้อมูลประกอบด้วยแอทริบิวต์ชนิดตัวเลขทั้งหมดทุกแอทริบิวต์ ดังตัวอย่างในรูปที่ 2(c) เราสามารถมองดาต้าอ็อบเจ็กต์แต่ละตัวได้ว่าเป็นจุดข้อมูล (หรือเวคเตอร์) ในปริภูมิเวกเตอร์หลายมิติ (multidimensional vector space) โดยที่แต่ละมิติในปริภูมิเวกเตอร์นั้นเป็นตัวแทนของแอทริบิวต์แต่ละตัวของดาต้าอ็อบเจ็กต์ เซตของดาต้าอ็อบเจ็กต์ก็จะสามารถตีความได้ว่าเป็นเมทริกซ์ ขนาด m แถว n หลัก เมื่อ m คือจำนวนดาต้าอ็อบเจ็กต์ในชุดข้อมูล และ n คือจำนวนแอทริบิวต์ในชุดข้อมูล เราจะเรียกเมทริกซ์นี้ว่า เมทริกซ์ข้อมูล (data matrix) การดำเนินการเมทริกซ์มาตรฐานต่าง ๆ สามารถนำมาใช้ในการแปลงและจัดการข้อมูลชนิดนี้ได้ ตัวอย่างการทำเหมืองข้อมูลที่มักใช้ data matrix ในการประมวลผล เช่นระบบผู้แนะนำ (recommender system) เป็นต้น
- **The Sparse Data Matrix** คือดาต้าเมทริกซ์ชนิดพิเศษซึ่งแอทริบิวต์ทุกตัวมีชนิดเดียวกันทั้งหมดและเป็นแอทริบิวต์แบบไม่สมมาตร ตัวอย่างของ sparse data matrix ที่พบบ่อยในการทำเหมืองข้อมูล เช่น document-term matrix ดังรูปที่ 2(d) แต่ละแถวของดาต้าเมทริกซ์คือตัวแทนของเอกสารแต่ละเอกสาร ส่วนคอลัมน์แต่ละคอลัมน์ของดาต้าเมทริกซ์จะแทนจำนวนครั้งที่คำศัพท์แต่ละคำที่ปรากฏในเอกสาร ในทางปฏิบัติเราจะเก็บเฉพาะข้อมูลในคอลัมน์ที่มีมากกว่าศูนย์ เท่านั้น

ข้อมูลกราฟ (Graph-Based Data)

กราฟเป็นโครงสร้างข้อมูล สำหรับแสดงความสัมพันธ์ระหว่างดาต้าอ็อบเจ็กต์ในชุดข้อมูล เรามักใช้กราฟในการทำเหมืองข้อมูลในสองกรณี คือ (1) เมื่อต้องการแสดงความสัมพันธ์ระหว่างดาต้าอ็อบเจ็กต์ เช่น ข้อมูลเว็บกราฟที่ประกอบด้วยโหนดคือเว็บเพจ และลิงก์คือ hyperlink (โดยใช้แทก a href) ระหว่างเว็บเพจแต่ละเพจ ดังรูปที่ 3(a) และ (2) เมื่อดาต้าอ็อบเจ็กต์มีโครงสร้างแบบกราฟ เช่น สารประกอบเบนซิน ซึ่งประกอบด้วยอะตอมของคาร์บอน (สีดำ) และไฮโดรเจน (สีเทา) เชื่อมต่อกันด้วยพันธะเคมีเป็นโครงสร้างดังแสดงในรูปที่ 3(b)



(a) Linked web pages. (b) Benzene molecule.

รูปที่ 3 ข้อมูลกราฟชนิดต่าง ๆ

ข้อมูลแบบมีลำดับ (Ordered Data)

ข้อมูลบางชนิดประกอบด้วยแอทริบิวต์ที่มีความสัมพันธ์กันเชิงเวลาหรือเชิงพื้นที่ เช่น sequential data, sequence data, time series data, spatial data ดังแสดงในรูปที่ 4

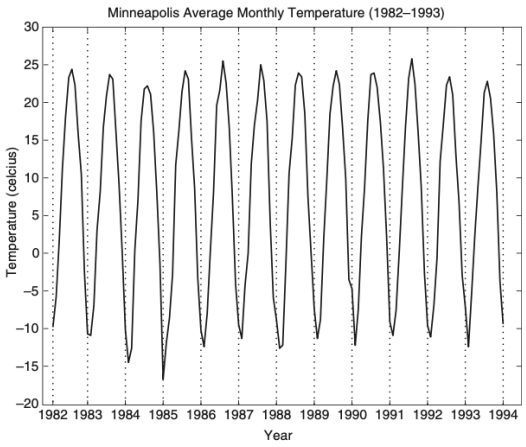
Time	Customer	Items Purchased
t1	C1	A, B
t2	C3	A, C
t2	C1	C, D
t3	C2	A, D
t4	C2	E
t5	C1	A, E

Customer	Time and Items Purchased
C1	(t1: A,B) (t2:C,D) (t5:A,E)
C2	(t3: A, D) (t4: E)
C3	(t2: A, C)

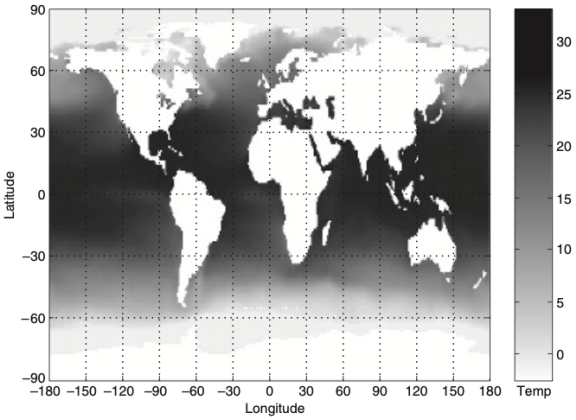
(a) Sequential transaction data.

```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

(b) Genomic sequence data.



(c) Temperature time series.



(d) Spatial temperature data.

รูปที่ 4 ข้อมูลเชิงลำดับชนิดต่าง ๆ

- **Sequential Transaction Data** คือข้อมูลทรานแซกชันที่มีแอทริบิวต์เวลาเพิ่มเข้ามา ดังตัวอย่างในรูปที่ 4(a) ประกอบด้วย เวลาทั้งหมด 5 จุดเวลาได้แก่ t1, t2, t3, t4, t5 ลูกค้า 3 เจ้า คือ C1, C2, C3 และสินค้าทั้งหมด 5 ไอเท็ม ได้แก่ A, B, C, D, E ข้อมูลเรคอร์ดแรกในตารางแสดงรายการสินค้าที่ถูกซื้อโดย C1 ณ เวลา t1 คือ A, B ข้อมูลเรคอร์ที่สองแสดงข้อมูลการซื้อสินค้า A, C โดย C3 ณ เวลา t2 เป็นต้น
- **Sequence Data** ข้อมูลลำดับประกอบด้วยดาต้าเซตซึ่งเป็นลำดับของดาต้าไอเท็ม เช่น ลำดับคำศัพท์ ลำดับของการซื้อสินค้า ข้อมูลลำดับคล้ายกันกับข้อมูลเชิงลำดับ (sequential data) มากแต่ต่างกันที่ข้อมูลลำดับไม่มีเวลา หรือ time stamps กำกับในแต่ละเรคอร์ด แต่ตำแหน่งที่ของดาต้าไอเท็มในชุดข้อมูลจะมีความสำคัญ เช่นข้อมูล DNA

ของสิ่งมีชีวิตสามารถแทนได้ด้วยลำดับนิวคลีโอไทด์ที่ประกอบด้วยสัญลักษณ์ 4 ตัว คือ A (adenine) C (cytosine) T (thymine) G (guanine) ดังแสดงในรูปที่ 4(b) ลำดับของโมเลกุลในข้อมูล DNA มีความสำคัญและจำเป็นต่อการวิเคราะห์ข้อมูล

- **Time Series Data** คือข้อมูลแบบมีลำดับชนิดพิเศษ ซึ่งแต่ละเรคอร์ดคืออนุกรมเวลาของผลการวัดค่าแอทริบิวต์ในแต่ละจุดเวลา เช่น ดาต้าเซตข้อมูลราคาหุ้นในแต่ละวัน หรือในรูปที่ 4(c) แสดงอนุกรมเวลาของอุณหภูมิเฉลี่ยรายเดือนในมินนีแอโพลิสระหว่าง ค.ศ 1982 ถึง 1994 ในการทำเหมืองข้อมูล คุณลักษณะสำคัญอย่างหนึ่งของข้อมูลเชิงเวลาที่ต้องคำนึงถึงคือ สหสัมพันธ์อัตโนมัติเชิงเวลา หรือ temporal autocorrelation ซึ่งหมายถึงการที่ค่าที่วัดได้สองค่าในเวลาที่ใกล้เคียงกัน มักจะมีค่าที่คล้ายคึงกันมาก
- **Spatial and Spatio-Temporal Data** ข้อมูลเชิงพื้นที่และข้อมูลเชิงพื้นที่-เวลา เช่น ข้อมูลสภาพอากาศ (ความชื้น อุณหภูมิ ความกดอากาศ) ที่เก็บรวบรวมมาจากพื้นที่ที่อยู่ในตำแหน่งต่าง ๆ กัน ในแต่ละช่วงเวลา คุณสมบัติที่สำคัญของข้อมูลเชิงพื้นที่ คือ สหสัมพันธ์อัตโนมัติเชิงพื้นที่ (**spatial autocorrelation**) ซึ่งหมายถึงการที่อ็อบเจ็กต์ที่อยู่ใกล้เคียงกันในทางกายภาพมักจะมีคุณสมบัติที่คล้ายคลึงกัน ดาต้าเซตสำหรับงานทางวิทยาศาสตร์และวิศวกรรมศาสตร์มักจะเป็นข้อมูลเชิงพื้นที่ หรือ ข้อมูลเชิงพื้นที่-เวลา เนื่องจากข้อมูลทางวิทยาศาสตร์มักเป็นผลมาจากการวัดหรือผลลัพธ์การคำนวณของโมเดล ที่ถูกบันทึกจากตำแหน่งต่าง ๆ ณ ช่วงเวลาต่าง ๆ กัน เช่น รูปที่ 4(d) แสดงแผนภาพข้อมูลอุณหภูมิ ณ ตำแหน่งละติจูดและลองจิจูดต่าง ๆ กันบนพื้นผิวโลก โดยสีเข้มหมายถึงอุณหภูมิสูง ส่วนสีอ่อนหมายถึงอุณหภูมิที่ต่ำกว่า

อัลกอริทึมการทำเหมืองข้อมูลส่วนใหญ่ ออกแบบมาสำหรับข้อมูลเรคอร์ด ในกรณีที่ข้อมูลถูกจัดเก็บอยู่ในรูปแบบที่ไม่ใช่ข้อมูลเรคอร์ด เราจะต้องแปลงข้อมูลเหล่านั้นให้อยู่ในรูปแบบของข้อมูลเรคอร์ดก่อน ด้วยเทคนิคการดึงคุณลักษณะ (feature extraction techniques) ซึ่งการแปลงข้อมูลให้อยู่ในรูปแบบเรคอร์ดอาจทำให้สูญเสียคุณสมบัติบางอย่างไป เช่น การแปลงข้อมูลเชิงพื้นที่-เวลาให้เป็นข้อมูลเรคอร์ดอาจทำให้ spatial autocorrelation และ temporal autocorrelation ระหว่างดาต้าอ็อบเจ็กต์ต่าง ๆ ไม่สามารถเห็นได้ชัดเจน การทำเหมืองข้อมูลในกรณีนี้จำเป็นจะต้องคำนึงถึง คุณสมบัติดังกล่าวด้วย แม้ว่าจะไม่ปรากฏชัดเมื่อแปลงข้อมูลให้อยู่ในรูปแบบเรคอร์ดก็ตาม

2.2 คุณภาพข้อมูล (Data Quality)

ในการทำเหมืองข้อมูล การแก้ไขปัญหาคุณภาพข้อมูลเป็นสิ่งที่หลีกเลี่ยงไม่ได้ เทคนิคในการจัดการกับคุณภาพของข้อมูลแบ่งได้เป็น 2 กลุ่มคือ (1) เทคนิคสำหรับการตรวจจับและการแก้ไขปัญหาคุณภาพข้อมูล (เรียกว่า data cleaning หรือการทำความสะอาดข้อมูล) และ (2) การใช้อัลกอทริทึมการทำเหมืองข้อมูลที่ทนทานต่อข้อมูลคุณภาพต่ำ

2.2.1 ความผิดพลาดจากการวัดและการเก็บข้อมูล

ปัญหาคุณภาพข้อมูลเกิดได้จากความผิดพลาดของมนุษย์ ข้อจำกัดของอุปกรณ์วัด หรือข้อบกพร่องในกระบวนการเก็บข้อมูล ในหัวข้อนี้เราจะพิจารณาถึงปัญหาที่เกิดจากการวัด ได้แก่ ข้อมูลรบกวน (noise), ข้อมูลเทียม (artifacts), ความลำเอียง (bias), ความเที่ยงตรง (precision), และ ความแม่นยำ (accuracy) และปัญหาที่เกิดจากทั้งการวัดและกระบวนการเก็บข้อมูล ได้แก่

ค่าผิดปกติ (outlier), ค่าที่ขาดหายและค่าที่ไม่สอดคล้อง (missing and inconsistent values), และค่าซ้ำ (duplicate values)

- **ความผิดพลาดจากการวัด (measurement error)** หมายถึงปัญหาที่เกิดจากกระบวนการวัด ปัญหาที่พบบ่อยคือค่าที่บันทึกแตกต่างไปจากค่าที่แท้จริง สำหรับแอทริบิวต์แบบค่าต่อเนื่อง (continuous attributes) ผลต่างระว่างค่าที่วัดได้กับค่าที่แท้จริงเรียกว่า **error**
- **ความผิดพลาดจากการเก็บข้อมูล (data collection error)** หมายถึงความผิดพลาดที่เกิดจากการไม่บันทึกข้อมูลบางค่า หรือการใส่ข้อมูลที่ไม่เหมาะสมเข้ามาในชุดข้อมูล

ข้อมูลรบกวน (Noise) และ ข้อมูลเทียม (Artifacts) Noise คือองค์ประกอบแบบสุ่ม (random component) ของความผิดพลาดจากการวัด ซึ่งมักเกิดจากการบิดเบือนของค่า หรือ การปรากฏของข้อมูลปลอม เช่น รูปที่ 5 แสดงข้อมูลอนุกรมเวลาที่ถูกรบกวนด้วย ข้อมูลรบกวนแบบสุ่มทำให้เกิดการเปลี่ยนรูปร่างไป รูปที่ 6 แสดงข้อมูลเชิงพื้นที่หลังจากมีข้อมูลรบกวน (เครื่องหมาย +) เพิ่มเข้าไป การกำจัด noise ออกจากดาต้าเซตมักทำได้ยาก ดังนั้นการทำเหมืองข้อมูลจึงเน้นที่การสร้างอัลกอริทึมที่ทนทานต่อ noise เรียกว่า robust algorithms ซึ่งสามารถสร้างผลลัพธ์ที่ยอมรับได้แม้ว่าจะมี noise อยู่ในชุดข้อมูลก็ตาม สำหรับความผิดพลาดที่เกิดจากการบิดเบือนของข้อมูลอย่างเป็นระบบและกำหนดได้ (demeterministic distortions) เราจะเรียกความผิดพลาดนี้ว่า artifacts หรือข้อมูลเทียม

ความเที่ยงตรง (Precision), ความลำเอียง (Bias), และความแม่นยำ (Accuracy)

คำนิยาม 2.3 ความเที่ยงตรง (Precision). คือความใกล้เคียงกันของการวัดค่าของสิ่งเดียวกันซ้ำกันหลายๆ ครั้ง

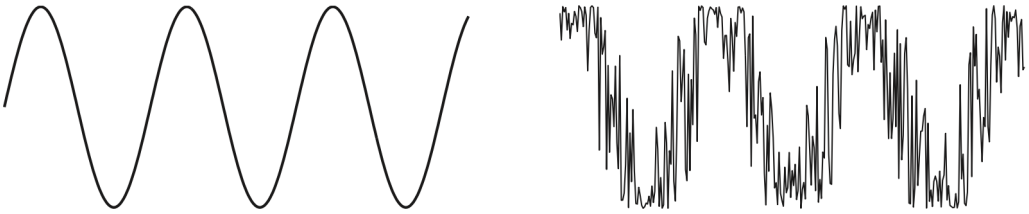
คำนิยาม 2.4 ความลำเอียง (Bias). คือการแปรผันอย่างเป็นระบบของการวัดจากค่าที่แท้จริงของสิ่งที่ถูกวัด

ความเที่ยงตรงมักจะวัดได้โดยใช้ค่าส่วนเบี่ยงเบนมาตรฐาน ส่วนความลำเอียงสามารถวัดได้โดยหาผลต่างระหว่างค่าเฉลี่ยของค่าที่วัดได้กับค่าที่แท้จริง

ตัวอย่างที่ 2.4 สมมติว่าเรามีก้อนน้ำหนักหนึ่งก้อนที่ทราบน้ำหนักที่แท้จริงคือ 1g การประเมินความเที่ยงตรงและความลำเอียงของตาชั่ง สามารถทำได้โดยการใช้ตาชั่งวัดน้ำหนักของก้อนน้ำหนักดังกล่าว 5 ครั้ง ซึ่งได้ค่าน้ำหนักในการชั่งแต่ละครั้งคือ {1.015, 0.990, 1.013, 1.001, 0.986} จากข้อมูลนี้ค่าเฉลี่ยของผลการวัดเท่ากับ 1.001 ดังนั้นค่า Bias จึงมีค่าเท่ากับ 1.001 – 1.000 = 0.001 และ Precision มีค่าเท่ากับส่วนเบี่ยงเบนมาตรฐาน ซึ่งก็คือ 0.013

คำนิยาม 2.5 ความแม่นยำ (Accuracy). คือความใกล้เคียงกันของค่าที่วัดได้กับค่าที่แท้จริงของสิ่งที่ต้องการวัด

ความแม่นยำของการวัดขึ้นกับความเที่ยงตรงและความลำเอียงในการวัด การบันทึกผลการวัดจะต้องบันทึกผลโดยคำนึงถึง เลขนัยสำคัญ (significant digits) กล่าวคือในการบันทึกผลจะต้องใช้จำนวนตัวเลขที่แทนผลการวัดหรือการคำนวณตามความเที่ยงตรงของข้อมูล เช่น ในการวัดความยาวของวัตถุโดยใช้ตลับเมตร ถ้าตลับเมตรมีสเกลที่ละเอียดที่สุดถึงหน่วยมิลลิเมตร เราก็ต้องบันทึกผลความยาววัตถุละเอียดที่สุดถึงระดับมิลลิเมตรที่ใกล้เคียงที่สุดเท่านั้น ความเที่ยงตรงของการวัดครั้งนี้คือ +/- 0.5 มิลลิเมตร



(a) Time series. (b) Time series with noise.

รูปที่ 5 Noise ในข้อมูลอนุกรมเวลา

(a) Three groups of points. (b) With noise points (+) added.

รูปที่ 6 Noise ในข้อมูลเชิงพื้นที่

ค่าผิดปกติ (Outliers)

ค่าผิดปกติ อาจหมายถึง (1) ดาต้าอ็อบเจ็กต์ ที่มีคุณลักษณะแตกต่างไปจากดาต้าอ็อบเจ็กต์อื่น ๆ ในดาต้าเซต หรือ (2) ค่าของแอทริบิวต์ที่ผิดปกติไปจากค่าที่ควรจะเป็น outliers ต่างจาก noise คือ outlier เป็นดาต้าอ็อบเจ็กต์จริงหรือเป็นค่าของแอทริบิวต์ของอ็อบเจ็กต์ที่เราสนใจศึกษา แตกต่างจาก noise ซึ่งเป็นสิ่งปลอมปนที่เราไม่ต้องการให้มีอยู่ในดาต้าเซต

ค่าที่ขาดหาย (Missing Values)

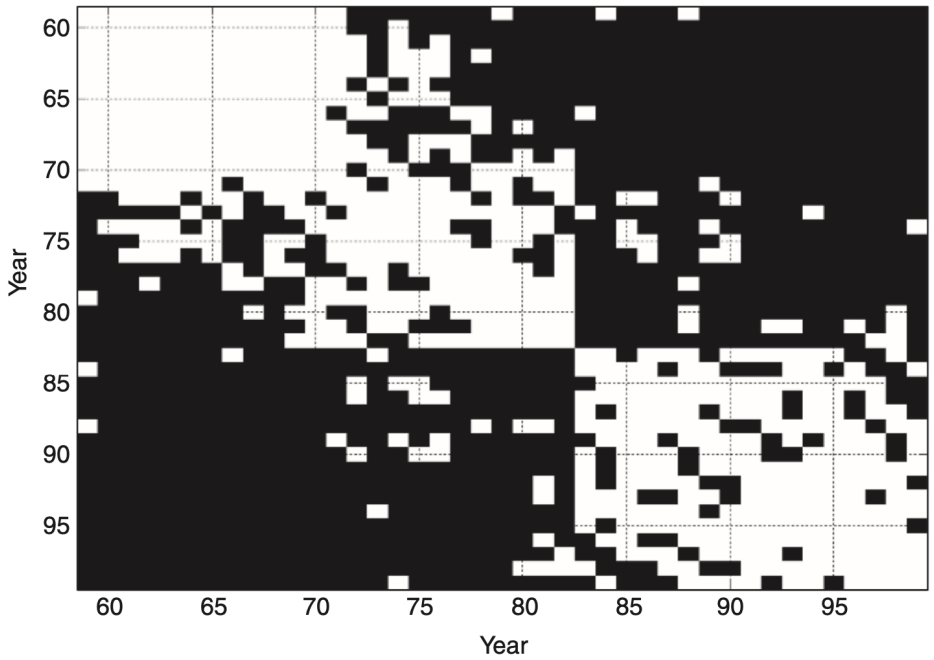
ข้อมูลหรือแอทริบิวต์ของข้อมูลอาจขาดหายไป เนื่องจากไม่ได้ถูกเก็บ (data not collected) เช่น ผู้ให้ข้อมูลปฏิเสธจะให้ข้อมูลเกี่ยวกับอายุและน้ำหนัก หรือบางกรณีอาจเกิดจากแอทริบิวต์ที่ไม่สามารถใช้ได้กับดาต้าอ็อบเจ็กต์ทุกตัว (data not applicable) การวิเคราะห์ข้อมูลจะต้องมีการจัดการ missing values ที่เหมาะสมซึ่งมีหลายแนวทาง แต่ละแนวทางเหมาะสมกับสถาณการณ์ที่ต่างกัน ดังต่อไปนี้คือ

- **การลบจุดข้อมูลหรือแอทริบิวต์ที่มีค่าขาดหายทิ้งไป** วิธีนี้เหมาะกับกรณีที่มีจำนวนของ missing values จำนวนน้อย และต้องทำด้วยความรอบคอบเนื่องจากข้อมูลที่ถูกลบทิ้งไปอาจมีความสำคัญต่อผลการวิเคราะห์ได้

- **การประมาณค่าที่ขาดหายไป** เช่นในข้อมูลอนุกรมเวลา เนื่องจากคุณสมบัติ temporal autocorrelation ทำให้เราสามารถประมาณค่าที่ขาดหายไปจากจุดข้อมูลที่อยู่ใกล้เคียงกันได้ โดยอาจใช้ค่าที่อยู่ใกล้ที่สุดหรือค่าเฉลี่ยของค่าที่อยู่ใกล้เคียงกันแทนค่าที่หายไปได้
- **ไม่นำค่าที่ขาดหายไปมาใช้ในการวิเคราะห์** เช่น ในการจัดกลุ่มข้อมูล เราสามารถดัดแปลงอัลกอริทึมให้เลือกใช้เฉพาะแอทริบิวต์ที่มีค่าครบถ้วนเท่านั้นในการคำนวณหาความคล้ายคลึงกันระหว่างดาต้าอ็อบเจ็กต์

ค่าที่ไม่สอดคล้อง (inconsistent values)

ตัวอย่างเช่น ข้อมูลรหัสไปรษณีย์ที่ไม่ตรงกันกับที่อยู่ที่ระบุไว้ในฟิลด์อื่น ๆ ค่าที่ไม่สอดคล้องจะต้องได้รับการจัดการก่อนเริ่มทำเหมืองข้อมูล หรือความไม่สอดคล้องกันที่เกิดจากการเปลี่ยนวิธีการที่ใช้ในการเก็บข้อมูล เช่น ข้อมูลอุณหภูมิที่พื้นผิวน้ำทะเล (Sea Surface Temperature: SST) ที่เก็บข้อมูลโดยใช้ทุ่นลอยระหว่างปี 1958-1982 และเก็บข้อมูลโดยใช้ดาวเทียมระหว่างปี 1983-1999 จากแผนภาพใน ที่แสดงสหสัมพันธ์ของข้อมูล SST ระหว่างแต่ละคู่ปี พบว่า ข้อมูลที่ได้จากแหล่งเดียวกัน จะมีสหสัมพันธ์เชิงบวก (สีขาว) ส่วนข้อมูลที่ได้จากคนละแหล่งกันจะมีสหสัมพันธ์เชิงลบ (สีดำ)



รูปที่ 7 Correlation ของข้อมูล SST ระหว่างแต่ละคู่ปี พื้นที่สีขาวแสดงถึงสหสัมพันธ์เชิงบวก ส่วนพื้นที่สีดำแสดงสหสัมพันธ์เชิงลบ

ข้อมูลซ้ำ (Duplicate Data)

ดาต้าเซตมักมีดาต้าอ็อบเจ็กต์ที่ซ้ำกันหรือเกือบซ้ำกันรวมอยู่ด้วย เราเรียกกระบวนการในการจัดการกับปัญหาที่เกิดจากข้อมูลซ้ำว่า **deduplication** ซึ่งมักประกอบด้วยการตรวจหาข้อมูลซ้ำและการผสานรวมข้อมูลซ้ำเข้าด้วยกัน ข้อควรระวังในการทำ deduplication คือ (1) ต้องระวังไม่ให้เกิดกรณีการผสานรวมข้อมูลที่ไม่ใช่ข้อมูลซ้ำเข้าด้วยกัน (2) ส่วนมากเมื่อข้อมูลซ้ำกัน มักจะมีค่าของแอทริบิวต์บางตัวที่แตกต่างกันหรือไม่สอดคล้องกัน การผสานรวมจึงต้องทำด้วยความรอบคอบ

2.2.2 ปัญหาคุณภาพที่เกี่ยวกับการประยุกต์ใช้งาน

ข้อมูลที่มีคุณภาพสูง คือข้อมูลที่เหมาะสำหรับการประยุกต์ใช้งานที่ต้องการ คุณภาพข้อมูลจากมุมมองของการประยุกต์ใช้งานมีหลายแง่มุม แต่ปัญหาทั่วไปของคุณภาพข้อมูลในแง่การนำไปใช้งาน มี 3 อย่าง คือ ความทันสมัย (timeliness), ความเกี่ยวข้อง (relevance), และ คำอธิบายข้อมูล (knowledge about the Data)

- **Timeliness** ข้อมูลที่นำมาใช้ในการทำเหมืองข้อมูลจะต้องทันสมัย หรือถูกเก็บมาในช่วงเวลาที่เหมาะสม
- **Relevance** จะต้องมีข้อมูลทุกชิ้นที่จำเป็นต้องใช้ในการวิเคราะห์อยู่ในดาต้าเซต เช่นการสร้างโมเดลเพื่อทำนายอัตราการเกิดอุบัติเหตุในการขับขี่รถยนต์ หากผู้เก็บข้อมูลไม่เก็บข้อมูลอายุและเพศของผู้ขับขี่ อาจจะส่งผลให้โมเดลที่ได้มีความแม่นยำจำกัดได้ ความท้าทายอีกอย่างหนึ่งของการสร้างชุดข้อมูลที่มีข้อมูลครบถ้วนมักเกิดจากปัญหาในเชิงสถิติที่เรียกว่า ความลำเอียงของการสุ่มตัวอย่าง (**sampling bias**) ซึ่งเกิดขึ้นเมื่อการสุ่มตัวอย่างที่มีสัดส่วนของอ็อบเจ็กต์แต่ละชนิดไม่ตรงกันกับสัดส่วนของอ็อบเจ็กต์แต่ละชนิดที่มีอยู่จริงในประชากร (population) ความลำเอียงของการสุ่มตัวอย่าง มักจะทำให้โมเดลเกิดมีผิดพลาดสูงเมื่อนำไปใช้งานจริง
- **Knowledge about the Data** คุณภาพของเอกสารคู่มือของชุดข้อมูลสามารถช่วยหรือเป็นอุปสรรคต่อการวิเคราะห์ข้อมูลต่อไปได้ สิ่งที่ควรระบุไว้ในเอกสารคู่มือของชุดข้อมูล เช่น แหล่งที่มาของข้อมูล วิธีการเก็บข้อมูล กฎเกณฑ์ที่ใช้ในการเก็บข้อมูล คุณสมบัติของแอทริบิวต์แต่ละตัว จำนวนข้อมูลที่เก็บได้ เป็นต้น

2.3 การเตรียมข้อมูลก่อนการประมวลผล (Data Preprocessing)

เทคนิคการเตรียมข้อมูลก่อนการทำเหมืองข้อมูล ที่สำคัญที่จะกล่าวถึงในหัวข้อนี้ แบ่งเป็นสองกลุ่มคือ กลุ่มแรกเป็นเทคนิคที่ใช้สำหรับลดจำนวนข้อมูลหรือมิติของข้อมูล ได้แก่ aggregation, sampling, dimensionality reduction, feature subset selection และกลุ่มที่สองเป็นเทคนิคที่ใช้สำหรับสร้างหรือเปลี่ยนแปลงแอทริบิวต์ ได้แก่ feature creation, discretization and binarization, variable transformation

2.3.1 Aggregation

Aggregation คือการรวมดาต้าอ็อบเจ็กต์หลาย ๆ ตัวให้เหลือเพียงอ็อบเจ็กต์เดียว เช่น การรวมรายการขายสินค้ารายวันแต่ละชนิดของร้านแต่ละสาขาให้เหลือเพียงสรุปยอดขายรายวันของแต่ละสาขา จะช่วยลดจำนวนเรคอร์ดข้อมูลที่ต้องประมวลผลลงไปได้เป็นจำนวนมาก การรวมแอทริบิวต์เชิงปริมาณ เช่น ยอดขาย ทำได้โดยการหาผลรวมหรือค่าเฉลี่ย สำหรับแอทริบิวต์เชิงคุณภาพ เช่น ประเภทสินค้า อาจรวมได้โดยการตัดค่าทิ้งหรือการสรุปโดยใช้โครงสร้างของประเภทข้อมูลเช่น สินค้าไอโฟน สามารถสรุปได้ว่าเป็นสินค้าในกลุ่ม smart phones และ สินค้าในกลุ่ม smart phones สามารถสรุปเป็นสินค้าในกลุ่ม electronics เป็นต้น

การรวมข้อมูลด้วยการลดจำนวนข้อมูลที่เป็นไปได้ เช่น จาก 365 วัน ไปเป็น 12 เดือน หรือการสรุปแอทริบิวต์จากประเภทย่อย ๆ ไปสู่ประเภทที่กว้างขึ้น เช่นจาก smart phones ไปเป็น electronics เป็นเทคนิคการรวมข้อมูลที่มักใช้ใน การประมวลผลเชิงวิเคราะห์แบบออนไลน์ (Online Analytical Processing: OLAP)

ประโยชน์ของการรวมข้อมูล มีหลายประการ คือ ประการแรกดาต้าเซตมีขนาดเล็กลงทำให้เวลาและหน่วยความจำที่ต้องใช้ในการประมวลผลน้อยลง ส่งผลให้เราสามารถใช้อัลกอริทึมการทำเหมืองข้อมูลที่มีความซับซ้อนมากขึ้นได้ ประการที่สอง การรวมข้อมูลสามารถใช้สำหรับการเปลี่ยนระดับขอบเขตของข้อมูล (เช่น ยอดขายรายปีของแต่ละสาขาให้มุมมองระดับสูง ส่วนยอดขายรายวันของแต่ละสาขาให้มุมมองระดับต่ำ เป็นต้น) ประการสุดท้ายเป็นเหตุผลทางสถิติที่ว่าปริมาณที่ได้จากการทำ aggregation เช่น ค่าเฉลี่ยหรือผลรวม จะมีความแปรปรวนน้อยลง ข้อเสียของการทำ aggregation คือการสูญเสียรายละเอียดของข้อมูล เช่น หลังจากสรุปยอดขายจากรายวันไปเป็นรายเดือนแล้ว เราไม่สามารถทราบได้ว่าวันใดคือวันที่มียอดขายสูงที่สุดหรือต่ำที่สุดได้ เป็นต้น

2.3.2 Sampling

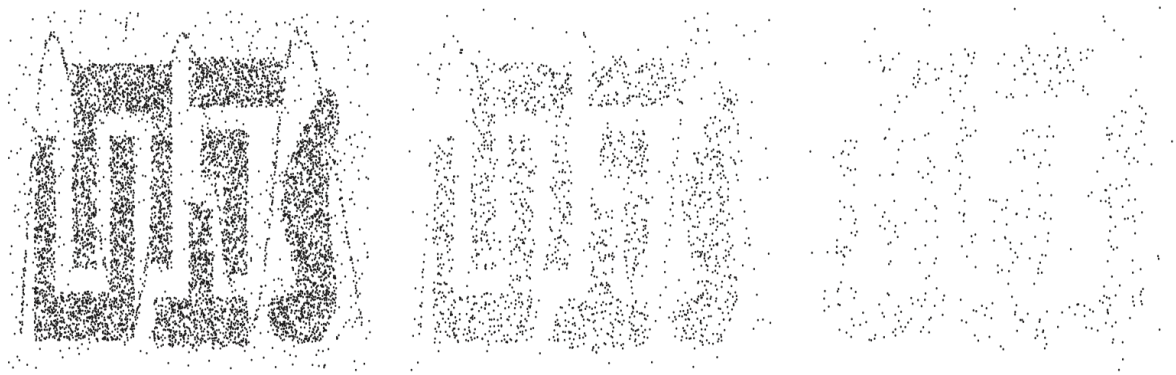
การสร้างกลุ่มตัวอย่าง (Sampling) คือวิธีการคัดเลือกดาต้าอ็อบเจ็กต์บางตัวเพื่อนำไปใช้ในการวิเคราะห์ การสร้างกลุ่มตัวอย่างเป็นเทคนิคที่ใช้มาอย่างยาวนานในวิชาสถิติทั้งก่อนและหลังการวิเคราะห์ข้อมูล นักสถิติสร้างกลุ่มตัวอย่างเพื่อรวบรวมข้อมูลที่สนใจจากกลุ่มประชากร เนื่องจากการเก็บข้อมูลจากทุกตัวอย่างในประชากรใช้เวลาและค่าใช้จ่ายสูงมากหรือไม่สามารถทำได้ ส่วนนักทำเหมืองข้อมูลใช้การสร้างกลุ่มตัวอย่างเพื่อลดปริมาณข้อมูลที่จะต้องประมวลผล เนื่องจากข้อมูลมีปริมาณมากจนไม่สามารถประมวลผลได้ด้วยทรัพยากรที่มีอยู่ได้อย่างมีประสิทธิภาพ หลักการสร้างกลุ่มตัวอย่างที่สำคัญก็คือ ข้อมูลกลุ่มตัวอย่างจะสามารถนำไปใช้ทำเหมืองข้อมูลได้ดีพอกันกับการทำเหมืองข้อมูลโดยใช้ข้อมูลเดิมเมื่อกลุ่มตัวอย่างที่ได้เป็นตัวแทนของข้อมูลเดิม (representative)

วิธีสร้างกลุ่มตัวอย่าง

- การสุ่มตัวอย่าง (random sampling) มี 2 รูปแบบคือ การเลือกตัวอย่างโดยไม่มีการคืนที่ (sampling without replacement) และการเลือกตัวอย่างโดยมีการคืนที่ (sampling with replacement) ในการเลือกตัวอย่างโดยมีการคืนที่ ดาต้าอ็อบเจ็กต์สามารถถูกเลือกได้มากกว่าหนึ่งครั้ง หากสัดส่วนของขนาดกลุ่มตัวอย่างต่อขนาดของดาต้าเซตมีค่าน้อย ผลลัพธ์ที่ได้จากทั้งสองวิธีการจะไม่แตกต่างกันมากนัก ข้อมูลตัวอย่างที่ได้มากจากการเลือกตัวอย่างโดยมีการคืนที่ จะนำไปใช้ในการวิเคราะห์ได้ง่ายกว่าเนื่องจากความน่าจะเป็นของการที่ดาต้าอ็อบเจ็กต์หนึ่งจะถูกเลือกจะมีค่าคงที่ตลอดกระบวนการเลือกตัวอย่าง
- การสุ่มตัวอย่างแบบชั้นภูมิ (stratified sampling) ใช้เมื่อข้อมูลประกอบด้วยข้อมูลแตกต่างกันหลายชนิด แต่ละชนิดมีจำนวนที่แตกต่างกันมาก มีขั้นตอน คือ เริ่มจากการแบ่งข้อมูลออกเป็นกลุ่มที่มีลักษณะเหมือนกัน (homogenous) จากนั้นจึงสุ่มตัวอย่างเพื่อให้ได้จำนวนกลุ่มตัวอย่าง ตามสัดส่วนของขนาดกลุ่มแต่ละกลุ่มในข้อมูลเดิม
- การสุ่มตัวอย่างแบบก้าวหน้า (progressive sampling) ใช้เมื่อการหาขนาดข้อมูลที่เหมาะสมทำได้ยาก โดยมีวิธีการคือ เริ่มจากสร้างข้อมูลตัวอย่างขนาดเล็ก จากนั้นค่อย ๆ เพิ่มขนาดของข้อมูลตัวอย่างไปจนกระทั่งข้อมูลที่ได้มีความเหมาะสมกับการประยุกต์ใช้งาน ตัวอย่างเช่น การเลือกตัวอย่างเพื่อนำข้อมูลที่ได้ไปใช้สร้างโมเดลทำนายค่าเราสามารถหาขนาดข้อมูลตัวอย่างที่เหมาะสมได้โดย เริ่มจากการสร้างโมเดลด้วยข้อมูลตัวยอย่างขนาดเล็กแล้ววัดความแม่นยำของโมเดลที่สร้างขึ้น จากนั้นค่อย ๆ เพิ่มขนาดข้อมูลให้ใหญ่ขึ้นเรื่อย ๆ จนกระทั่งพบว่าความแม่นยำของโมเดลทำนายค่ามีค่าลดลง

ขนาดที่เหมาะสมของการเลือกตัวอย่าง ยิ่งขนาดของการเลือกตัวอย่างใหญ่ ก็จะยิ่งเพิ่มความน่าจะเป็นที่จะได้ข้อมูลที่เป็นตัวแทนที่ดี (representative) ให้สูงยิ่งขึ้น แต่ในขณะเดียวกันก็จะทำให้ประโยชน์ที่จะได้รับจากการเลือกตัวอย่าง คือการลด

ปริมาณข้อมูลที่ต้องประมวลผล ลงไปด้วย ในทางกลับกันการเลือกตัวอย่างที่มีขนาดเล็กเกินไปจะเกิดผลเสียคือทำให้สูญเสียรูปแบบที่สำคัญหรือทำให้เกิดรูปแบบที่ไม่เป็นจริงขึ้นได้ เช่น รูปที่ 8 แสดงผลกระทบของขนาดของการเลือกตัวอย่างที่มีต่อการสูญเสียรูปแบบที่แฝงในชุดข้อมูล จากรูปที่ 8(a) ข้อมูลเดิมที่มีจำนวน 8,000 จุดเมื่อทำการเลือกตัวอย่างขนาด 2,000 จุดได้ผลลัพธ์เป็นข้อมูลดังรูปที่ 8(b) ซึ่งจะเห็นได้ว่ายังมีโครงสร้างที่คล้ายคลึงกันกับข้อมูลเดิมอยู่พอสมควร แต่เมื่อลดขนาดการเลือกตัวอย่างลงเหลือ 500 จุด พบว่าโครงสร้างในข้อมูลส่วนใหญ่ได้สูญเสียไปดังรูปที่ 8(c)



(a) 8000 points (b) 2000 points (c) 500 points

รูปที่ 8 การสูญเสียโครงสร้างในข้อมูลกับขนาดของการเลือกตัวอย่าง

2.3.3 Dimensionality Reduction

อัลกอริทึมการทำเหมืองข้อมูลมักจะทำงานได้ดีขึ้นเมื่อจำนวนมิติของข้อมูล (จำนวนแอทริบิวต์) มีจำนวนน้อยลง การลดจำนวนมิติข้อมูล (dimensionality reduction) กำจัดคุณลักษณะหรือฟีเจอร์ที่ไม่เกี่ยวข้อง และลดข้อมูลรบกวน (noise) และช่วยลดปัญหาที่เกิดจาก the curse of dimensionality ได้ นอกจากนี้การลดจำนวนมิติข้อมูลยังช่วยให้สามารถแสดงผลข้อมูลด้วย visualization แบบต่าง ๆ ได้ง่ายขึ้น โมเดลมีความซับซ้อนน้อยลง เวลาและหน่วยความจำที่ใช้ในการทำเหมืองข้อมูลลดลง คำว่า dimensionality reduction เป็นคำที่ใช้เรียกเทคนิคที่ลดจำนวนมิติข้อมูล โดยการสร้างแอทริบิวต์ใหม่ขึ้นแอทริบิวต์เก่าที่มีอยู่เดิม ส่วนเทคนิคที่ลดจำนวนมิติข้อมูลโดยการเลือกซับเซตของแอทริบิวต์ที่มีอยู่เดิมจะเรียกว่า feature subset selection หรือ feature selection ซึ่งจะได้กล่าวถึงในหัวข้อต่อไป

เทคนิคการลดจำนวนมิติข้อมูลที่เป็นที่นิยม เช่น Principal Component Analysis (PCA) ซึ่งเป็นเทคนิคของพีชคณิตเชิงเส้น สำหรับสร้างแอทริบิวต์ใหม่จากผลรวมเชิงเส้นของแอทริบิวต์ที่มีอยู่เดิม โดยมิติใหม่ที่ได้แต่ละมิติจะตั้งฉากกัน (orthogonal) และมิติแต่ละมิติจะเรียงตัวในทิศทางที่มีความผันผวนของข้อมูลสูงที่สุด

The Curse of Dimensionality คือปรากฏการณ์ที่การวิเคราะห์ข้อมูลทำได้ยากขึ้นมากเมื่อจำนวนมิติ (หรือจำนวนแอทริบิวต์) ของข้อมูลสูงขึ้น ปรากฏการณ์นี้มีสาเหตุมาจากการที่ข้อมูลเกิดการกระจายตัวสูงขึ้นเมื่อมีจำนวนมิติมากขึ้น ซึ่งจะทำให้ปริมาณข้อมูลที่ต้องใช้เพื่อเป็นตัวแทนของรูปแบบต่าง ๆ สูงขึ้น ส่งผลให้ประสิทธิภาพของอัลกอริทึมการจำแนกประเภทและการจัดกลุ่มลดลง ตามจำนวนมิติข้อมูลที่เพิ่มขึ้น

2.3.4 Feature Subset Selection

คือเทคนิคการลดจำนวนมิติข้อมูลโดยการคัดเลือกซับเซตของแอทริบิวต์ ให้เหลือแต่แอทริบิวต์ที่ไม่ซ้ำกัน (non-redundant features) และเกี่ยวข้องกับการวิเคราะห์ข้อมูล (relevant features) วิธีการที่ง่ายที่สุดสำหรับการคัดเลือกซับเซตฟีเจอร์คือการแจกแจงและทดสอบซับเซตที่เป็นไปได้ทั้งหมดของชุดข้อมูล ซึ่งหากชุดข้อมูลมีฟีเจอร์หรือแอทริบิวต์ n ตัว จำนวนซับเซตที่เป็นไปได้ทั้งหมดของชุดข้อมูลจะมีค่าเท่ากับ 2^n ดังนั้นวิธีการนี้จึงใช้ไม่ได้ในทางปฏิบัติ เนื่องจากจำนวนแอทริบิวต์ที่พบในสถาณการณ์จริงมักมีจำนวนมาก ซึ่งบางครั้งอาจมีจำนวนเป็นร้อยหรือเป็นพันแอทริบิวต์

เทคนิคการคัดเลือกซับเซตฟีเจอร์ (feature subset selection techniques) ที่ใช้กันทั่วไป มี 3 ประเภท คือ embedded approaches, filter approaches, และ wrapper approaches

Embedded approaches การคัดเลือกฟีเจอร์เป็นขั้นตอนหนึ่งในกระบวนการทำเหมืองข้อมูล เช่น ในระหว่างการสร้างต้นไม้ตัดสินใจ (decision tree) อัลกอริทึมสร้างต้นไม้ตัดสินใจจะเลือกว่าแอทริบิวต์ใดบ้างที่จะถูกนำมาใช้เป็นเกณฑ์ในการตัดสินใจ

Filter approaches ฟีเจอร์จะถูกคัดเลือกก่อนเริ่มประมวลผลอัลกอริทึมการทำเหมืองข้อมูล ด้วยเทคนิควิธีการที่เป็นอิสระจากวิธีการทำเหมืองข้อมูลที่นำมาใช้ เช่น การคัดเลือกฟีเจอร์โดยใช้ค่าสหสัมพันธ์ของฟีเจอร์แต่ละคู่เป็นเกณฑ์คัดกรองฟีเจอร์ที่ซ้ำกัน (redundant features) ออกไป

Wrapper approaches เป็นวิธีการที่ใช้อัลกอริทึมการทำเหมืองข้อมูลเป็นเหมือนกล่องดำสำหรับทดสอบเพื่อค้นหาซับเซตของแอทริบิวต์ที่ดีที่สุด วิธีการนี้คล้ายกันกับการแจกแจงซับเซตที่เป็นไปได้ทั้งหมด ต่างกันที่ wrapper approaches ไม่จำเป็นต้องทดสอบทุกซับเซตที่เป็นไปได้

การหาค่าถ่วงน้ำหนักของฟีเจอร์ (Feature Weighting)

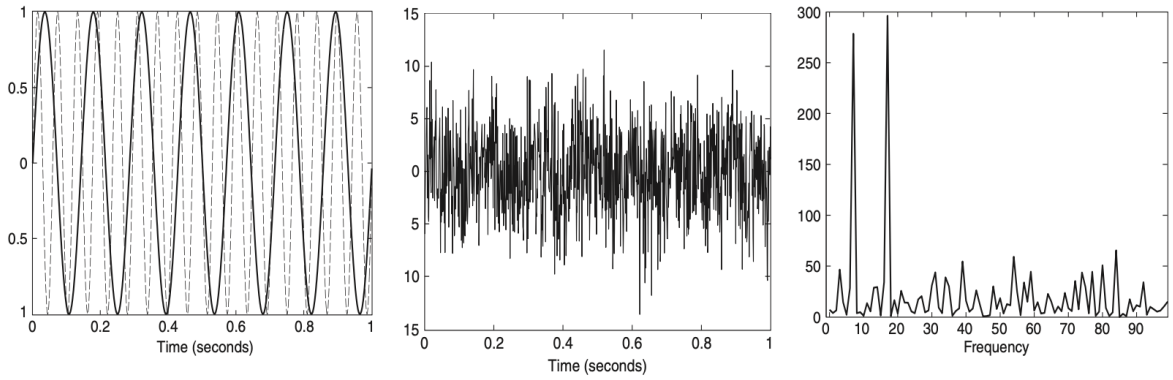
แทนที่จะเก็บแอทริบิวต์บางตัวและตัดแอทริบิวต์ที่เหลือทิ้ง อีกแนวทางหนึ่งคือการกำหนดค่าถ่วงน้ำหนักให้กับฟีเจอร์แต่ละตัว โดยฟีเจอร์ที่สำคัญกว่าจะมีค่าถ่วงน้ำหนักสูงกว่า การกำหนดค่าถ่วงน้ำหนักอาจทำได้โดยใช้ความรู้เกี่ยวกับความสำคัญของฟีเจอร์แต่ละตัวของผู้เชี่ยวชาญ หรืออาจกำหนดโดยอัตโนมัติ ด้วยอัลกอริทึม เช่น support vector machine เป็นต้น

2.3.5 Feature Creation

วิธีการสร้างฟีเจอร์ ที่จะกล่าวถึงในหัวข้อนี้ มีสองวิธีการคือ การสกัดฟีเจอร์ (feature extraction) และ การแมปข้อมูลไปยังปริภูมิใหม่ (mapping the data to a new space)

กาสกัดฟีเจอร์ คือการสร้างฟีเจอร์ใหม่จากฟีเจอร์ดั้งเดิม การสกัดฟีเจอร์เป็นเทคนิคที่ขึ้นกับโดเมนข้อมูล เช่น การสกัดฟีเจอร์ของรูปภาพจากข้อมูลดิบคือ พิกเซลข้อมูล ทำได้โดยการใช้เทคนิคการประมวลผลรูปภาพเช่น การตรวจจับเส้นขอบ (edge detection) เป็นต้น

การแมปข้อมูลไปยังปริภูมิใหม่ที่ทำให้สามารถวิเคราะห์ข้อมูลได้ง่ายกว่า เช่น การแปลงข้อมูลอนุกรมเวลาที่อยู่ในโดเมนเวลาไปอยู่ในโดเมนความถี่โดยใช้การแปลงฟูเรียร์ (Fourier transform) แล้วทำการวิเคราะห์สเปกตรัมกำลังจะทำให้สามารถแยกข้อมูลจริงออกจากข้อมูลรบกวนได้ ดังเช่นตัวอย่างในรูปที่ 9



(a) Two time series. (b) Noisy time series. (c) Power spectrum.

รูปที่ 9 การประยุกต์ใช้ Fourier transform เพื่อระบุความถี่มูลฐานในข้อมูลอนุกรมเวลา

2.3.6 Discretization and Binarization

อัลกอริทึมการจำแนกประเภท (classification algorithms) บางอัลกอริทึมใช้งานได้กับแอทริบิวต์เชิงคุณภาพ (categorical) เท่านั้น ส่วนอัลกอริทึมการวิเคราะห์ความสัมพันธ์ (association analysis) ต้องใช้กับแอทริบิวต์แบบไบนารี (binary) เท่านั้น ฉะนั้นหากดาต้าเซตของเรามีแอทริบิวต์แบบค่าต่อเนื่อง (continuous attributes) เราจำเป็นต้องแปลงแอทริบิวต์เหล่านั้นให้อยู่ในรูปแบบของแอทริบิวต์เชิงคุณภาพก่อนโดยการทำ **discretization** หรือบางกรณีเราอาจจำเป็นต้องแปลงแอทริบิวแบบค่าต่อเนื่องและเชิงคุณภาพไปเป็นแอทริบิวต์แบบไบนารีโดยการทำ **binarization**

Binarization

ถ้าแอทริบิวต์ประกอบด้วยค่าที่เป็นไปได้ทั้งหมด m ค่า แล้ว เราสามารถแปลงแอทริบิวต์ดังกล่าวไปเป็นเลขฐานสองที่มีขนาดอย่างน้อย $\log_2$ m บิต เช่น แอทริบิวต์ movie_ratings ประกอบด้วยค่าที่เป็นไปได้ทั้งหมด 5 ค่าคือ awful, poor, OK, good, great เราสามารถแปลงแอทริบิวต์ movie_ratings ไปเป็นไบนารีที่มีขนาด 3 บิตได้โดย

1) กำหนดค่าจำนวนเต็มระหว่าง 0-2^3 ให้กับค่าที่เป็นไปได้แต่ละค่า เช่น
   awful=0, poor=1. OK=2, good=3, great=4
2) แปลงค่าจำนวนเต็มไปเป็นเลขฐานสอง :
   awful=000, poor=001, OK=010, good=011, great=100

อีกวิธีการหนึ่งในการทำ binarization เรียกว่า hot encoding ซึ่งทำได้โดยการแทนค่าที่เป็นไปได้แต่ละค่าโดยใช้บิตข้อมูลแต่ละตำแหน่ง เช่น แอทริบิวต์ movie_ratings ประกอบด้วยค่าที่เป็นไปได้ 5 ค่า ดังนั้น จะต้องใช้ไบนารีขนาด 5 บิตในการแทนข้อมูล โดยกำหนดให้ตำแหน่งบิตแต่ละตำแหน่งแทนค่าที่เป็นไปได้แต่ละค่า ดังนั้น จะได้ว่า awful=10000, poor=01000, OK=00100, good=00010, great=00001

Discretization

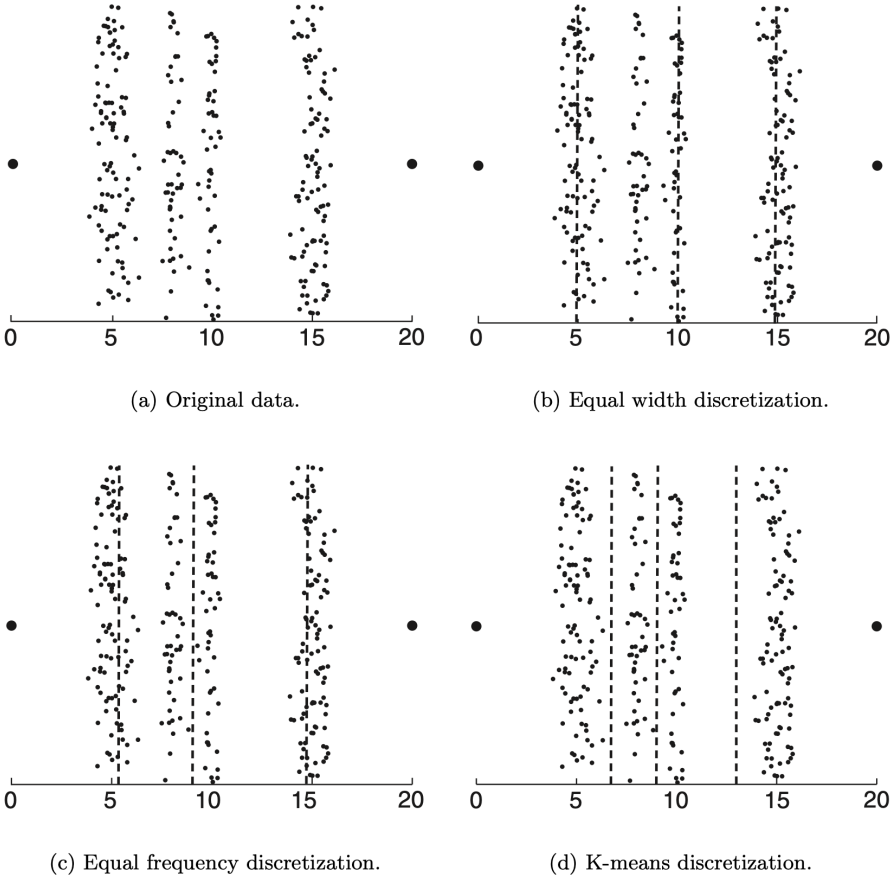
การแปลงข้อมูลเป็นค่าไม่ต่อเนื่อง (Discretization) มักใช้เพื่อแปลงแอทริบิวต์แบบต่อเนื่อง (continuous attributes) ไปเป็นแอทริบิวต์เชิงคุณภาพ (categorical attributes) เพื่อใช้ในการทำเหมืองข้อมูลเพื่อจำแนกประเภทข้อมูลหรือวิเคราะห์ความสัมพันธ์

การแปลงข้อมูลเป็นค่าไม่ต่อเนื่อง มี 2 ขั้นตอน คือ

1) กำหนดจำนวนค่าที่เป็นไปได้ (n) จากนั้นเรียงลำดับข้อมูลจากน้อยไปมาก และแบ่งข้อมูลออกเป็น n ช่วงค่า โดยการกำหนดจุดแบ่ง (split points) จำนวน n-1 จุด
2) ค่าของแอทริบิวต์แบบต่อเนื่องที่อยู่ในช่วงเดียวกัน จะถูกแปลงเป็นแอทริบิวต์เชิงคุณภาพค่าเดียวกัน

จากขั้นตอนข้างต้น จะเห็นได้ว่าการแปลงข้อมูลเป็นค่าไม่ต่อเนื่อง ก็คือการกำหนดจำนวนจุดแบ่งข้อมูล และตำแหน่งของจุดแบ่งข้อมูลแต่ละจุดนั่นเอง ซึ่งมีเทคนิคพื้นฐานที่ใช้บ่อย 3 เทคนิค (ดัง) คือ

- Equal width discretization คือการแบ่งแต่ละช่วงข้อมูลให้มีความกว้างเท่ากัน
- Equal frequency discretization คือการแบ่งแต่ละช่วงข้อมูลให้มีจำนวนจุดข้อมูล (frequency) เท่ากัน
- K-means discretization คือการแบ่งแต่ละช่วงข้อมูลโดยใช้การจัดกลุ่มแบบ k-means



(a) Original data. (b) Equal width discretization.

(c) Equal frequency discretization. (d) K-means discretization.

รูปที่ 10 เทคนิคการแปลงข้อมูลเป็นค่าไม่ต่อเนื่อง (discretization techniques)

2.3.7 Variable Transformation

variable transformation หรือ attribute transformation หมายถึงการแปลงค่าแอทริบิวต์ที่ถูกนำไปใช้กับค่าทุกค่าของตัวแปร เช่น การแปลงค่าแอทริบิวต์โดยใช้ฟังก์ชันค่าสัมบูรณ์ (absolute function) เมื่อเราสนใจเฉพาะขนาดของข้อมูล การแปลงค่าตัวแปรที่สำคัญที่จะกล่าวถึงในที่นี้มี 2 ประเภทคือ การแปลงเชิงฟังก์ชันอย่างง่าย (simple functional transformation) และการแปลงให้อยู่ในรูปแบบมาตรฐาน (normalization)

การแปลงเชิงฟังก์ชันอย่างง่าย คือการแทนค่าแอทริบิวต์ x ด้วยค่าที่คำนวณได้จากฟังก์ชันทางคณิตศาสตร์ เช่น x^k, log x, e^x, 1/x, sin x, sqrt(x) หรือ |x| ในการทำเหมืองข้อมูลเรามักจำเป็นต้องแปลงค่าข้อมูลให้อยู่ในช่วงที่เหมาะสมกับการประมวลผล เช่น ในการทำเหมืองข้อมูลเพื่อตรวจจับการบุกรุกในเครือข่าย เราอาจสนใจวิเคราะห์ขนาดของข้อมูลที่ถูกส่งมาบนเครือข่ายของแต่ละเซสชัน ซึ่งโดยปกติจะมีช่วงที่กว้างมาก ตั้งแต่ 1 ไบต์ ไปจนถึง 1 กิกะไบต์ การบีบช่วงข้อมูลให้แคบลงด้วยฟังก์ชันลอการิทึม จะช่วยให้การทำความเข้าใจและวิเคราะห์ข้อมูลทำได้ง่ายขึ้นมาก การแปลงฟังก์ชันต้องทำด้วยความระมัดระวังเนื่องจากการแปลงค่าด้วยฟังก์ชันบางชนิดทำให้คุณสมบัติของข้อมูลเปลี่ยนไปจากเดิม เช่น ฟังก์ชัน 1/x จะทำให้ลำดับของข้อมูลเปลี่ยนไปจากเดิม

การแปลงให้อยู่ในรูปแบบมาตรฐาน มีเป้าหมายคือการทำให้ค่าทั้งหมดของแอทริบิวต์มีคุณสมบัติที่ต้องการ เช่น การทำข้อมูลให้เป็นมาตรฐาน เพื่อให้แอทริบิวต์ x มีค่าเฉลี่ยเป็น 0 และค่าส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 ทำได้โดยใช้สูตร $x' = (x - \overline{x})/s_x$ เมื่อ $\overline{x}$ คือค่าเฉลี่ย และ s_x คือค่าส่วนเบี่ยงเบนมาตรฐานของข้อมูล การแปลงข้อมูลให้อยู่ในรูปแบบมาตรฐานช่วยป้องกันไม่ให้แอทริบิวต์ที่มีค่าสูงมีอิทธิพลเหนือแอทริบิวต์ตัวอื่น ซึ่งจะส่งผลกระทบต่อผลลัพธ์ของการทำเหมืองข้อมูล ในกรณีที่ข้อมูลประกอบด้วยค่าผิดปกติ (outliers) จำนวนมาก การแปลงข้อมูลให้อยู่ในรูปมาตรฐานโดยใช้สูตรคำนวณข้างต้นอาจไม่เหมาะสม เนื่องจากค่าเฉลี่ยและค่าส่วนเบี่ยงเบนมาตรฐานมีความอ่อนไหวมากต่อค่าผิดปกติ (outliers) วิธีการแก้ไขทำได้โดยแทนที่ค่าเฉลี่ยในสูตรด้วยค่ามัธยฐาน (median) และแทนที่ค่าส่วนเบี่ยงเบนมาตรฐานด้วยค่าส่วนเบี่ยงเบนมาตรฐานสัมบูรณ์ (absolute standard deviation) ซึ่งคำนวณได้จาก สูตร $\sigma_A = \sum_{i=1}^{m} |x_i - \mu|$ เมื่อ x_i ค่าตัวที่ i ของแอทริบิวต์ x , m คือจำนวนของดาต้าอ็อบเจ็กต์ทั้งหมด และ μ คือค่าเฉลี่ยหรือค่ามัธยฐานของแอทริบิวต์ x

2.4 การวัดความเหมือนและความแตกต่าง (Measures of Similarity and Dissimilarity)

ความเหมือนและความแตกต่าง ถูกนำไปใช้ในเทคนิคการทำเหมืองข้อมูลหลายเทคนิค เช่น การจัดกลุ่ม การจำแนกประเภทโดยใช้เพื่อนบ้านที่ใกล้เรียง (nearest neighbor classfication) และการตรวจจับค่าผิดปกติ (anomaly detection)

2.4.1 พื้นฐาน

คำนิยาม 2.6 Similarity (ความคล้ายคลึง หรือ ความละม้าย). similarity ระหว่างอ็อบเจ็กต์สองอ็อบเจ็กต์ คือปริมาณเชิงตัวเลขที่แสดงถึงระดับความคล้ายคลึงกันของอ็อบเจ็กต์ทั้งสอง ยิ่งอ็อบเจ็กต์คู่นั้นมีความคล้ายคลึงกันมากเท่าใด ค่า similarity ก็จะยิ่งมีค่ามาก โดยปกติค่า similarity จะมีค่าอยู่ระหว่าง 0 (ไม่เหมือนกันเลย) ถึง 1 (เหมือนกันอย่างสมบูรณ์)

คำนิยาม 2.7 Dissimilarity (ความแตกต่าง). dissimilarity ระหว่างอ็อบเจ็กต์สองอ็อบเจ็กต์ คือปริมาณเชิงตัวเลขที่แสดงถึงระดับความแตกต่างกันของอ็อบเจ็กต์ทั้งสอง dissimilarity จะมีค่าน้อยเมื่ออ็อบเจ็กต์มีความคล้ายคลึงกันมาก **distance (ระยะทาง)** คือมาตรวัดความแตกต่างชนิดหนึ่ง โดยปกติ dissimilarity จะมีค่าอยู่ระหว่าง 0 (ไม่แตกต่างกันเลย) ถึง 1 (แตกต่างกันอย่างสมบูรณ์) หรือระหว่าง 0 (ไม่แตกต่างกันเลย) ถึง ∞ (แตกต่างกันอย่างสมบูรณ์)

การแปลงค่าระหว่าง similarity และ dissimilarity สามารถทำได้โดยใช้ฟังก์ชันลดทางเดียว (monotonic decreasing function) เช่น $s = 1/(d+1)$, $s = e^{-d}$ เป็นต้น

Proximity (ความใกล้เคียง) เป็นคำที่ใช้หมายถึง similarity หรือ dissimilarity ระหว่างอ็อบเจ็กต์ สามารถคำนวณค่าได้จากฟังก์ชันของ similarity หรือ dissimilarity ระหว่างแอทริบิวต์แต่ละตัวของอ็อบเจ็กต์ทั้งสอง

2.4.2 การหาค่า Similarity และ Dissimilarity ระหว่างแอทริบิวต์เดี่ยว

การคำนวณหาค่า similarity และ dissimilarity ระหว่างแอทริบิวต์แต่ละชนิดสรุปได้ดังตารางที่ 2.3

ตารางที่ 2.3. similarity และ dissimilarity สำหรับแอทริบิวต์เดี่ยว (simple attribute)

Attribute Type	**Dissimilarity**	**Similarity**
Nominal	$d = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$	$s = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y \end{cases}$
Ordinal	$d = \lvert x - y \rvert / (n-1)$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - d$
Interval or Ratio	$d = \lvert x - y \rvert$	$s = -d,\ s = \frac{1}{1+d},\ s = e^{-d},$ $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

2.4.3 Dissimilarity ระหว่างดาต้าอ็อบเจ็กต์

Euclidean distance d ระหว่างดาต้าอ็อบเจ็กต์สองอ็อบเจ็กต์ **x** และ **y** สามารถคำนวณได้จากสูตร

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2},$$

เมื่อ x_k , y_k คือ แอทริบิวต์ตัวที่ **k** ของอ็อบเจ็กต์ **x** และ **y** ตามลำดับ และ n คือจำนวนแอทริบิวต์ทั้งหมด

ตัวอย่างที่ 2.5 กำหนด x และ y เป็นดาต้าอ็อบเจ็กต์ที่ประกอบด้วย ratio attributes สี่ตัว มีค่าดังตาราง จงคำนวณหา Euclidean distance ระหว่าง x และ y

attributes	x	y
a1	0	3
a2	2	1
a3	-1	5

$d(x, y) = \sqrt{(\sum_{k=1 \text{ to } 3} (x_k - y_k)^2)}$

$= \sqrt{[(0-3)^2 + (2-1)^2 + (-1-5)^2]}$

$= \sqrt{[\ 9 + 1 + 36]}$

$= \sqrt{46} \approx 6.78$

Minkowski distance คือมาตรวัดระยะทางที่มีรูปแบบทั่วไป ดังสูตร

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{k=1}^{n} |x_k - y_k|^r \right)^{1/r},$$

เราสามารถเลือกวิธีการวัดระยะทางได้โดยการกำหนดค่าพารามิเตอร์ r เช่น

- r = 1. City block (Manhattan, taxicab, L1 norm) distance.
- r = 2. Euclidean distance (L2 norm).

คุณสมบัติของ Euclidean distance

ถ้า d(x, y) คือระยะทางระหว่างดาต้าอ็อบเจ็กต์สองอ็อบเจ็กต์ x และ y แล้ว คุณสมบัติต่อไปนี้จะเป็นจริง:

1. **Positivity**

   (ก) d(x,y) >= 0 สำหรับทุกค่าของ x และ y

   (ข) d(x,y) = 0 ก็ต่อเมื่อ x = y

2. **Symmetry**

   d(x,y) = d(y,x) สำหรับทุกค่าของ x และ y

3. **Triangle Inequality**

   d(x,z) <= d(x,y) + d(y,z) สำหรับทุกดาต้าอ็อบเจ็กต์ x, y และ z

มาตรวัดใดที่มีคุณสมบัติครบทั้งสามข้อข้างต้นคือ Positivity, Symmetry, Triangle Inequality จะเรียกว่าเมตริก (metrics)

ตัวอย่างที่ 2.6 กำหนดเซต A = {1,2,3,4} และเซต B = {2,3,4} จงคำนวณหา d(A,B) ตามสูตรที่กำหนดให้ต่อไปนี้ และตรวจสอบว่า สูตรแต่ละสูตรคือเมตริกหรือไม่

(ก) d(A,B) = size(A – B)

(ข) d(A,B) = size(A – B) + size(B – A)

(ก) A – B = {1}, size(A – B) = 1 ดังนั้น d(A,B) = 1

d(A,B) = size(A – B) ไม่ใช่เมตริก เพราะขาดคุณสมบัติ Positiviy ข้อที่ 2 , Symmetry, และ Triangle inequality

(ข) A – B = {1}, size(A – B) = 1

B – A = {}, size (B – A) = 0

ดังนั้น d(A, B) = 1 + 0 = 1

d(A,B) = size(A – B) + size(B – A) เป็นเมตริก

2.4.4 Similarity ระหว่างดาต้าอ็อบเจ็กต์

ถ้า s(x, y) คือความคล้ายคลึงกันระหว่างดาต้าอ็อบเจ็กต์ x และ y แล้ว s(x, y) จะมีคุณสมบัติดังต่อไปนี้

1. s(x, y) = 1 ก็ต่อเมื่อ x = y. (0 <= s <= 1)
2. s(x, y) = s(y, x) สำหรับทุกค่าของ x และ y (Symmetry)

แม้ว่า Similarity จะไม่มีคุณสมบัติ triangle inequality แต่เราสามารถปรับเปลี่ยน similarity บางชนิด เช่น cosine และ Jaccard similarity measures ให้กลายเป็นระยะทางเมตริก (metric distance) ได้อย่างง่ายดาย

2.4.5 ตัวอย่างของ Proximity Measures

Similarity Measures สำหรับข้อมูลไบนารี (Similarity Coefficients)

กำหนดให้ x และ y เป็นดาต้าอ็อบเจ็กต์ที่ประกอบด้วยไบนารีแอทริบิวต์ n ตัว การเปรียบเทียบ x และ y จะทำให้เราสามารถหาค่าความถี่ได้ 4 ตัวดังนี้

f_{00} = จำนวนของแอทริบิวต์ที่ x = 0 และ y = 0

f_{01} = จำนวนของแอทริบิวต์ที่ x = 0 และ y = 1

f_{10} = จำนวนของแอทริบิวต์ที่ x = 1 และ y = 0

f_{11} = จำนวนของแอทริบิวต์ที่ x = 1 และ y = 1

- **Simple Matching Coefficient (SMC)**

$$SMC = \frac{\text{number of matching attribute values}}{\text{number of attributes}} = \frac{f_{11} + f_{00}}{f_{01} + f_{10} + f_{11} + f_{00}}.$$

- **Jaccard Coefficient** เหมาะกับกรณีที่แอทริบิวต์เป็นแบบ asymmetric binary attributes

$$J = \frac{\text{number of matching presences}}{\text{number of attributes not involved in 00 matches}} = \frac{f_{11}}{f_{01} + f_{10} + f_{11}}.$$

ตัวอย่างที่ 2.7 กำหนดดาต้าอ็อบเจ็กต์ x และ y ซึ่งประกอบด้วยแอทริบิวต์แบบไบนารีมีค่าดังนี้คือ

x = (1,0,0,0,0,0,0,0,0,0)

y = (0,0,0,0,0,0,1,0,0,1)

f_{00} = 7, f_{01} = 2, f_{10} = 1, f_{11} = 0

SMC = (0 + 7) / (2+1+0+7) = 7/10 = 0.7

J = 0 / (2 + 1 + 0) = 0

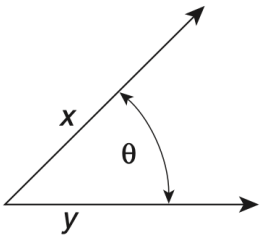
จะเห็นได้ว่า การวัดความคล้ายคลึงด้วย SMC จะให้ค่าสูงมาก แต่การวัดโดยใช้ Jaccard Coefficient กลับมีค่าเท่ากับศูนย์ซึ่งหมายความว่า x และ y ไม่เหมือนกันเลย ที่เป็นเช่นนี้เนื่องจาก Jaccard Coefficient จะไม่สนใจกรณีที่แอทริบิวต์ของ x และ y เป็น 0 ทั้งคู่นั่นเอง ดังนั้น Jaccard Coefficent จึงนิยมใช้เมื่อเราไม่สนใจกรณีที่ไบนารีแอทริบิวต์มีเป็นศูนย์หรือเมื่อแอทริบิวต์มีชนิดเป็น asymmetric binary attributes นั่นเอง

- **Cosine Similarity** เป็นมาตรวัดความคล้ายคลึงกันที่นิยมใช้มากในระบบการค้นคืนสารสนเทศ (information retrieval system) มีสูตรดังนี้คือ

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \, \|\mathbf{y}\|}$$

เมื่อ <x,y> คือ inner product ของเวคเตอร์ x และ y , || x || คือความยาวของเวคเตอร์ x, || y || คือความยาวของเวคเตอร์ y

ในทางเรขาคณิตค่าของ cos(x, y) ก็คือค่า cosine ของมุมระหว่างเวคเตอร์ x และเวคเตอร์ y ดังแสดงใน



รูปที่ 11 ความหมายในทางเรขาคณิตของ cosine similarity

ตัวอย่างที่ 2.8 กำหนดดาต้าอ็อบเจ็กต์ x และ y ที่ประกอบด้วยแอทริบิวต์ มีค่าดังนี้คือ

x = (3,2,0,5,0,0,0,2,0,0)

y = (1,0,0,0,0,0,0,1,0,2)

<x, y> = 3x1 + 2x0 + 0x0 + 5x0 + 0x0 + 0x0 + 0x0 + 2x1 + 0x0 + 0x2 = 3 + 2 = 5

|| x || = √ (3x3 + 2x2 + 0x0 + 5x5 + 0x0 + 0x0 + 0x0 + 2x2 + 0x0 + 0x0) = √ (9+4+25+4) = √42 ≈ 6.48

|| y || = √ (1x1 + 0x0 + 0x0 + 0x0 + 0x0 + 0x0 + 0x0 + 1x1 + 0x0 + 2x2) = √ (1 + 1 + 4) = √6 ≈ 2.45

cos(x, y) = <x,y> / (|| x || || y ||) = 5 / (6.48 x 2.45) = 0.31

- **Pearson's Correlation** คือตัววัดความสัมพันธ์เชิงเส้นระหว่างดาต้าอ็อบเจ็กต์สองอ็อบเจ็กต์ กำหนด x และ y ให้เป็นดาต้าอ็อบเจ็กต์สองอ็อบเจ็กต์ โดย x ประกอบด้วยแอทริบิวต์ x_1, x_2, ... x_n และ y ประกอบด้วยแอทริบิวต์ y_1, y_2, ..., y_n ค่าสหสัมพันธ์ Pearson's Correlation ระหว่าง x และ y จะมีค่าระหว่าง -1 ถึง 1 และสามารถคำนวณได้จากสูตร

$$\text{corr}(\mathbf{x}, \mathbf{y}) = \frac{\text{covariance}(\mathbf{x}, \mathbf{y})}{\text{standard\_deviation}(\mathbf{x}) \times \text{standard\_deviation}(\mathbf{y})} = \frac{s_{xy}}{s_x \; s_y},$$

เมื่อ

$$\text{covariance}(\mathbf{x},\mathbf{y}) = s_{xy} = \frac{1}{n-1}\sum_{k=1}^{n}(x_k - \overline{x})(y_k - \overline{y})$$

$$\begin{aligned}\text{standard\_deviation}(\mathbf{x}) &= s_x = \sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(x_k - \overline{x})^2} \\ \text{standard\_deviation}(\mathbf{y}) &= s_y = \sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(y_k - \overline{y})^2}\end{aligned}$$

$$\begin{aligned}\overline{x} &= \frac{1}{n}\sum_{k=1}^{n} x_k \text{ is the mean of } \mathbf{x} \\ \overline{y} &= \frac{1}{n}\sum_{k=1}^{n} y_k \text{ is the mean of } \mathbf{y}\end{aligned}$$

ตัวอย่างที่ 2.9 กำหนดดาต้าอ็อบเจ็กต์ x และ y ที่ประกอบด้วยแอทริบิวต์มีค่าดังนี้คือ

x = (-3, 6, 0, 3, -6)

y = (1, -2, 0, -1, 2)

Mean(x) = 0, Mean(y) = 0,

S_x = √ [(1/4)[(-3-0)^2 + (6-0)^2 + (0-0)^2 + (3-0)^2 + (-6-0)^2]] = 4.74

S_y = √ [(1/4)[(1-0)^2 + (-2-0)^2 + (0-0)^2 + (-1-0)^2 + (2-0)^2]] = 1.58

S_{xy} = (1/4) [(-3-0)(1-0) + (6-0)(-2-0) + (0-0)(0-0) + (3-0)(-1-0) + (-6-0)(2-0)] = -7.5

Corr(x, y) = -7.5 / (4.75 x 1.58) = -0.999 ≈ -1 ($x_k = -3y_k$)

ตัวอย่างที่ 2.10 กำหนดดาต้าอ็อบเจ็กต์ x และ y ที่ประกอบด้วยแอทริบิวต์มีค่าดังนี้คือ

x = (3, -6, 0, -3, 6)

y = (1, -2, 0, -1, 2)

Mean(x) = 0, Mean(y) = 0,

S_x = √ [(1/4)[(3-0)^2 + (-6-0)^2 + (0-0)^2 + (-3-0)^2 + (6-0)^2]] = 4.74

S_y = √ [(1/4)[(1-0)^2 + (-2-0)^2 + (0-0)^2 + (-1-0)^2 + (2-0)^2]] = 1.58

S_{xy} = (1/4) [(3-0)(1-0) + (-6-0)(-2-0) + (0-0)(0-0) + (-3-0)(-1-0) + (6-0)(2-0)] = 7.5

Corr(x, y) = 7.5 / (4.75 x 1.58) = 0.999 ≈ 1 ($x_k = 3y_k$)

ตัวอย่างที่ 2.11 กำหนดดาต้าอ็อบเจ็กต์ x และ y ที่ประกอบด้วยแอทริบิวต์มีค่าดังนี้คือ

x = (-3, -2, -1, 0, 1, 2, 3) ; y = (9, 4, 1, 0, 1, 4, 9)

Mean(x) = 0, Mean(y) = 4

$S_x = \sqrt{[(1/6)[(-3-0)^2 + (-2-0)^2 + (-1-0)^2 + (0-0)^2 + (1-0)^2 + (2-0)^2 + (3-0)^2]]}$

$= \sqrt{[(1/6)[9 + 4 + 1 + 0 + 1 + 4 + 9] = 2.16}$

$S_y = \sqrt{[(1/6)[(9-4)^2 + (4-4)^2 + (1-4)^2 + (0-4)^2 + (1-4)^2 + (4-4)^2 + (9-4)^2]]}$

$= \sqrt{[(1/6)[25 + 0+ 9 + 16 + 9 + 0 + 25]]} = 3.74$

$S_{xy} = (1/6) [(-3-0)(9-4)+(-2-0)(4-4)+(-1-0)(1-4)+(0-0)(0-4)+(1-0)(1-4)+(2-0)(4-4)+(3-0)(9-4)]$

$= (1/6) [-15 + 0 + 3 + 0 - 3 + 0 + 15] = 0$

Corr(x, y) = 0 / (2.16 * 3.74) = 0

คุณสมบัติการไม่แปรผันตามการปรับขนาดและการเลื่อนตำแหน่ง

cosine similarity, correlation, และ Minkowski distance measures มีคุณสมบัติการไม่แปรผันตามการปรับขนาด (invariant to scaling) และการเลื่อนตำแหน่ง (invariant to translation) สรุปได้ดังตารางที่ 2.4

ตารางที่ 2.4. คุณสมบัติของ Cosine, Correlation และ Minkowski Distance

Property	Cosine	Correlation	Minkowski Distance
Invariant to scaling (multiplication)	Yes	Yes	No
Invariant to translation (addition)	No	Yes	No

ตัวอย่างที่ 2.12 กำหนดดาต้าอ็อบเจ็กต์ x และ y ซึ่งประกอบด้วยแอทริบิวต์มีค่าดังนี้คือ

x = (1,2,4,3,0,0,0)

y = (1,2,3,4,0,0,0)

$y_s = 2 * y = (2,4,6,8,0,0,0)$

$y_t = y + 5 = (6,7,8,9,5,5,5)$

<x, y> = (1x1+2x2+4x3+3x4+0x0+0x0+0x0) = 29

$\langle x, y_s \rangle = (1x2+2x4+4x6+3x8+0x0+0x0+0x0) = 58$

$\langle x, y_t \rangle = (1x6+2x7+4x8+3x9+0x5+0x5+0x5) = 79$

$\| x \| = \sqrt{(1^2 + 2^2 + 4^2 + 3^2 + 0^2 + 0^2 + 0^2)} = \sqrt{(1 + 4 + 16 + 9)} = 5.4772$

$\| y \| = \sqrt{(1^2 + 2^2 + 3^2 + 4^2 + 0^2 + 0^2 + 0^2)} = \sqrt{(1 + 4 + 9 + 16)} = 5.4772$

$|| y_s || = \sqrt{(2^2 + 4^2 + 6^2 + 8^2 + 0^2 + 0^2 + 0^2)} = \sqrt{(4 + 16 + 36 + 64)} = 10.9545$

$|| y_t || = \sqrt{(6^2 + 7^2 + 8^2 + 9^2 + 5^2 + 5^2 + 5^2)} = \sqrt{(36 + 49 + 64 + 81 + 25 + 25 + 25)} = 17.4642$

Cosine(x, y) = 29 / (5.4772x5.4772) = 0.9667

Cosine(x, y_s) = 58 / (5.4772x10.9545) = 0.9667

Cosine(x, y_t) = 79 / (5.4772x17.4642) = 0.8259

Corr(x,y) = 2.4524 / (1.6183x1.6183) = 0.9364

Corr(x, y_s) = 4.9048/(1.6183x3.2367) = 0.9364

Corr(x, y_t) = 2.4524/(1.6183x1.6183) = 0.9364

Euclidean(x, y) = $\sqrt{(1-1)^2 + (2-2)^2 + (4-3)^2 + (3-4)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} = \sqrt{2} = 1.4142$

Euclidean(x, y_s) = $\sqrt{(1-2)^2 + (2-4)^2 + (4-6)^2 + (3-8)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} = \sqrt{(1+4+4+25)} = 5.8310$

Euclidean(x, y_t) = $\sqrt{(1-6)^2 + (2-7)^2 + (4-8)^2 + (3-9)^2 + (0-5)^2 + (0-5)^2 + (0-5)^2} = 13.3041$

ค่าความละม้ายระหว่าง (x, y) (x, y_s) (x, y_t) สรุปได้ดังนี้คือ

วิธีการวัด	**(x, y)**	**(x, y_s)** **x กับ y ที่ถูกปรับขนาดเพิ่มสองเท่า**	**(x, y_t)** **x กับ y ที่ถูกเลื่อนไป 5 หน่วย**
Cosine	0.9667	0.9667	0.8259
Correlation	0.9364	0.9364	0.9364
Euclidean Distance	1.4142	5.8310	13.3041

จากตารางจะเห็นได้ว่า ค่า Cosine, Correlation, Euclidean distance ของ (x, y) (x, y_s) และ (x, y_t) จะมีคุณสมบัติการไม่แปรผันตามการปรับขนาดและการเลื่อนตำแหน่งตามที่สรุปไว้ในตารางที่ 2.4

ในการทำเหมืองข้อมูล เราจะต้องเลือกใช้ proximity measure ให้เหมาะสมโดยคำนึงถึงคุณสมบัติการไม่แปรผันตามการปรับขนาดและการเลื่อนตำแหน่ง เช่น

- Cosine similarity เหมาะกับการหาความคล้ายคลึงกันระหว่างเอกสารในระบบค้นคืนสารสนเทศเนื่องจากในงานการค้นคืนเอกสารเราต้องการตัววัดความคล้ายคลึงที่ไม่ผันแปรตามการปรับขนาด แต่ต้องอ่อนไหวต่อการเลื่อนตำแหน่ง (หรือการปรากฏของคำศัพท์ที่แตกต่างกัน)
- พิจารณาสถาณการณ์ที่เราต้องการเก็บข้อมูล อุณหภูมิ ในตำแหน่งต่าง ๆ กันเป็นเวลา 7 วัน โดยที่ข้อมูลที่เก็บมาได้แต่ละจุดอาจอยู่ในหน่วยองศาเซลเซียส องศาฟาเรนไฮต์ หรือ องศาเคลวิน เนื่องจาก การแปลงจากค่าอุณหภูมิในแต่ละหน่วยต้องมีทั้งการคูณ (ปรับขนาด) และการบวก (เลื่อนตำแหน่ง) proximity meaures ที่เหมาะสมในกรณีนี้จะต้องไม่ได้รับผลกระทบที่เกิดขึ้นจากการเปลี่ยนหน่วยการวัดอุณหภูมิ ซึ่งมีทั้งการปรับขนาดและการเลื่อนตำแหน่ง ดังนั้น Correlation จึงเป็น proximity measures ที่เหมาะสมที่สุด

- Euclidean distance เป็น proximity measures ที่ไม่มีคุณสมบัติ scaling invariant และ translation invariant ดังนั้นจึงเหมาะกับกรณีที่เราต้องการตรวจจับการเปลี่ยนแปลงของดาต้าอ็อบเจ็กต์ทั้งในด้านขนาด (scaling) และทิศทาง (translation) เช่น การวัดความแม่นยำของโมเดลการทำนายค่า เป็นต้น

2.4.6 Mutual Information

Mutual information เป็นวิธีการวัดความคล้ายคลึงกันระหว่างอ็อบเจ็กต์สองอ็อบเจ็กต์เช่นเดียวกันกับ correlation แต่ mutual information สามารถใช้คำนวณหาความคล้ายคลึงระหว่างอ็อบเจ็กต์ได้ แม้ว่าอ็อบเจ็กต์คู่นั้นจะมีความสัมพันธ์แบบไม่เชิงเส้นก็ตาม

Mutual information เป็นตัววัดความเป็นอิสระต่อกันระหว่างเซตสองเซต หากเซตสองเซตมีความเป็นอิสระต่อกัน ค่า mutual information จะเป็น 0 แต่ถ้าเซตสองเซตไม่เป็นอิสระต่อกัน (การทราบค่าของเซตหนึ่ง ทำให้เราทราบข้อมูลเพิ่มเติมเกี่ยวกับอีกเซตหนึ่งได้) ค่า mutual information จะมีค่าสูงที่สุด กำหนดให้ X และ Y เป็นดาต้าอ็อบเจ็กต์สองอ็อบเจ็กต์ โดย X ประกอบด้วยแอทริบิวต์ u_1, u_2, ..., u_m และ Y ประกอบด้วยแอทริบิวต์ v_1, v_2, ..., v_n แล้วค่า Entropy H(X), Entropy H(Y) และ Joint Entropy H(X,Y) สามารถคำนวณได้โดยสูตร

$$H(X) = -\sum_{j=1}^{m} P(X = u_j) \log_2 P(X = u_j)$$

$$H(Y) = -\sum_{k=1}^{n} P(Y = v_k) \log_2 P(Y = v_k)$$

$$H(X,Y) = -\sum_{j=1}^{m}\sum_{k=1}^{n} P(X = u_j, Y = v_k) \log_2 P(X = u_j, Y = v_k)$$

ค่า Mutual Information ของ X และ Y คือ I(X, Y) สามารถคำนวณได้จาก H(X), H(Y) และ H(X,Y) ดังนี้คือ

$$I(X,Y) = H(X) + H(Y) - H(X,Y)$$

ตัวอย่างที่ 2.13 กำหนดดาต้าอ็อบเจ็กต์ X และ Y ซึ่งประกอบด้วยแอทริบิวต์มีค่าดังนี้คือ

x = (-3, -2, -1, 0, 1, 2, 3)

y = (9, 4, 1, 0, 1, 4, 9)

เราสามารถคำนวณค่า Entropy H(X) และ H(Y) ได้ดังตารางต่อไปนี้

x_j	$P(\mathbf{x} = x_j)$	$-P(\mathbf{x} = x_j) \log_2 P(\mathbf{x} = x_j)$
-3	$1/7$	0.4011
-2	$1/7$	0.4011
-1	$1/7$	0.4011
0	$1/7$	0.4011
1	$1/7$	0.4011
2	$1/7$	0.4011
3	$1/7$	0.4011
$H(\mathbf{x})$		2.8074

y_k	$P(\mathbf{y} = y_k)$	$-P(\mathbf{y} = y_k) \log_2 (P(\mathbf{y} = y_k)$
9	$2/7$	0.5164
4	$2/7$	0.5164
1	$2/7$	0.5164
0	$1/7$	0.4011
$H(\mathbf{y})$		1.9502

ค่า Joint Entropy H(X, Y):

x_j	y_k	$P(\mathbf{x}=x_j,\mathbf{y}=x_k)$	$-P(\mathbf{x}=x_j,\mathbf{y}=x_k)\log_2 P(\mathbf{x}=x_j,\mathbf{y}=x_k)$
-3	9	$1/7$	0.4011
-2	4	$1/7$	0.4011
-1	1	$1/7$	0.4011
0	0	$1/7$	0.4011
1	1	$1/7$	0.4011
2	4	$1/7$	0.4011
3	9	$1/7$	0.4011
$H(\mathbf{x},\mathbf{y})$			2.8074

ดังนั้น ค่า mutual information ของ X และ Y จะมีค่าเท่ากับ

I(X,Y) = H(X) + H(Y) – H(X, Y) = 2.8074 + 1.9502 – 2.8074 = 1.9502 ซึ่งมีค่ามากกว่า 0 แสดงว่า X และ Y มีความสัมพันธ์กันค่อนข้างสูง (strongly related)

2.4.7 การเลือกมาตรวัดความใกล้เคียง (Proximity Measure) ที่เหมาะสม

ข้อสังเกตในการเลือกใช้ proximity measure สำหรับการทำเหมืองข้อมูล มีดังนี้

1) ควรเลือกใช้ proximity measure ที่เหมาะสมกับชนิดของข้อมูล
   - **Euclidean distance** เป็น proximity measure ที่เหมาะกับข้อมูลที่มีความหนาแน่น (dense) และแอทริบิวต์เป็นแบบค่าต่อเนื่อง (continuous attributes)
   - ในกรณีที่ข้อมูลมีความหนาแน่นต่ำมาก (sparse) และประกอบด้วยแอทริบิวต์แบบไม่สมมาตร (asymmetric attributes) ควรเลือกใช้ proximity measure ที่ไม่สนใจคู่ของแอทริบิวต์ที่ตรงกันแบบ 0-0 เช่น Cosine และ Jaccard
   - การคำนวณความคลึงของข้อมูลอนุกรมเวลา (time series data) ควรใช้ correlation เนื่องจากเป็น proximity measure ที่ไม่แปรผันตามการปรับขนาด (invariant to scaling) หรือการเลื่อนตำแหน่ง (invariant to translation)
2) หากมีปัญหาหรือกังวลเกี่ยวกับประสิทธิภาพในการคำนวณ ให้เลือกใช้ proximity measures ที่มีคุณสมบัติ triangle inequality ซึ่งสามารถนำไปใช้เพื่อลดจำนวนครั้งของการคำนวณค่าความใกล้เคียงได้
3) การเลือก proximity measures ที่เหมาะสมต้องเกิดจากการใคร่ครวญอย่างรอบคอบโดยคำนึงถึงวัตถุประสงค์ของการวัดความคล้ายคลึง และความรู้ความเข้าใจเกี่ยวกับข้อมูลอย่างถ่องแท้

สรุป

- ข้อมูลคือวัตถุดิบตั้งต้นของการทำเหมืองข้อมูล ดังนั้นผู้ทำเหมืองข้อมูลจะต้องมีความรู้ความเข้าใจเกี่ยวคุณสมบัติของข้อมูลประเภทต่าง ๆ
- ประเภทของแอทริบิวต์แบ่งออกได้เป็น แอทริบิวต์เชิงคุณภาพ (qualitative attributes) และ แอทริบิวต์เชิงปริมาณ (quantitative attributes)
    - แอทริบิวต์เชิงคุณภาพมี 2 ชนิด ได้แก่ nominal attributes และ ordinal attributes
    - แอทริบิวต์เชิงปริมาณมี 2 ชนิด ได้แก่ interval attributes และ ratio attributes
- ประเภทของแอทริบิวต์จะเป็นตัวกำหนดโอเปอเรชันทางคณิตศาสตร์ที่สามารถใช้กับข้อมูลนั้นได้ (ดังแสดงในตารางที่ 2.2)
- ดาต้าเซต แบ่งตามชนิดข้อมูลได้เป็น record data (record, transaction, matrix), graph data, ordered data (sequential transaction, sequence, time series, spatio-temporal data)
- ปัญหาคุณภาพข้อมูลที่เกิดจากการวัดและการเก็บข้อมูล ได้แก่ noise และ artifacts, outliers, missing values, inconsistent values, duplicate data
- ปัญหาคุณภาพข้อมูลที่เกี่ยวกับการนำไปใช้งาน ได้แก่ ความทันสมัย (timeliness), ความเกี่ยวข้อง (relevance), คำอธิบายข้อมูล (knowledge about the data)
- เทคนิคสำหรับลดปริมาณข้อมูลที่ใช้บ่อย ได้แก่ การสรุปรวม (Aggregation), การเลือกตัวอย่าง (Sampling), การลดจำนวนมิติ (dimensionality reduction), การเลือกซับเซตฟีเจอร์ (feature subset selection)
- เทคนิคสำหรับการเตรียมข้อมูลให้อยู่ในรูปแบบที่เหมาะสมกับการทำเหมืองข้อมูล ได้แก่ การสร้างฟีเจอร์ (feature creation), การแปลงข้อมูลให้เป็นข้อมูลไม่ต่อเนื่องและไบนารี (discretization and binarization), การแปลงตัวแปร (variable tranformation)
- ในการทำเหมืองข้อมูล เรามักจำเป็นต้องคำนวณหาความคล้ายคลึงและความแตกต่างของข้อมูล มาตรวัดความคล้ายคลึงและความแตกต่าง (proximity measures) มาตรฐานที่ใช้บ่อยในการทำเหมืองข้อมูล ได้แก่ Cosine, Jaccard, Correlation, Euclidean, และ Mutual Information

แบบฝึกหัด

1. กำหนดแอทริบิวต์ดังต่อไปนี้ จงจำแนกประเภทของแอทริบิวต์ว่าเป็นแอทริบิวต์เชิงคุณภาพ (qualitative) ชนิดใด (nominal, ordinal) หรือเป็นแอทริบิวต์เชิงปริมาณ (quantitative) ชนิดใด (interval, ratio) นอกจากนี้ให้จำแนกด้วยว่าแอทริบิวต์แต่ละตัวมีชนิดเป็น Binary, Discrete, หรือ Continuous
    (ก) เวลาในรูปแบบ AM หรือ PM
    (ข) ความสว่างที่วัดโดยมิเตอร์วัดแสง
    (ค) ความสว่างที่วัดจากความรู้สึกของมนุษย์

(ง) มุมที่วัดเป็นองศาระหว่าง 0 ถึง 360

(จ) เหรียญทอง เหรียญเงิน เหรียญทองแดง ของกีฬาโอลิมปิก

(ฉ) ความสูงจากระดับน้ำทะเล

(ช) จำนวนผู้ป่วยในโรงพยาบาลแห่งหนึ่ง

(ซ) เลข ISBN ของหนังสือ

(ฌ) ความหนาแน่นของสสารหน่วยเป็น กรัมต่อลูกบาศก์เซนติเมตร

(ญ) ระยะทางจากจุดศูนย์กลางของวิทยาเขตหน่วยเป็นเมตร

(ฎ) ชั้นยศของกองทัพ

(ฏ) ความสามารถในการส่งผ่านคลื่นแสง: opaque translucent transparent

2. จงยกตัวอย่างสถาณการณ์ที่ identification numbers (รหัสประจำตัว) น่าจะมีประโยชน์สำหรับการทำนาย

3. ปริมาณใดต่อไปนี้ที่มีคุณสมบัติ spatial autocorrelation : daily rainfall หรือ daily temperature และทำไมจึงเป็นเช่นนั้น

4. โปรแกรมเมอร์คนหนึ่งได้ออกแบบอัลกอริทึม k-nearest neighbors ดังนี้

Algorithm 2.1 Algorithm for finding K nearest neighbors.

1: **for** $i = 1$ to *number of data objects* **do**
2: Find the distances of the i^{th} object to all other objects.
3: Sort these distances in decreasing order.
(Keep track of which object is associated with each distance.)
4: **return** the objects associated with the first K distances of the sorted list
5: **end for**

(ก) จงอภิปรายว่าจะมีปัญหาอะไรเกิดขึ้นได้บ้างกับอัลกอริทึมนี้ ถ้าดาต้าเซตมีข้อมูลซ้ำ (duplicates)

(ข) จงเสนอวิธีการแก้ไขปัญหาที่เกิดจากการมีข้อมูลซ้ำซ้อนในดาต้าเซต

5. คำนวณค่า cosine, correlation, Jaccard และ Euclidean distance ของ ดาต้าอ็อบเจ็กต์ x และ y ดังต่อไปนี้

(ก) x = (1, 1, 1, 1), y = (2, 2, 2, 2)

(ข) x = (0, 1, 0, 1), y = (1, 0, 1, 0)

(ค) x = (0, -1, 0, 1), y = (1, 0, -1, 0)

6. คำนวณค่า Mutual information ของดาต้าอ็อบเจ็กต์ x และ y ดังต่อไปนี้

(ก) x = (-7, -2, 1, 0, 1, 2), y = (9, 4, 1, 0, 4, 1)

(ข) x = (1,1,1,1), y = (2,2,2,2)

เอกสารอ้างอิง

[1] Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar. “Introduction to Data Mining”. Pearson, 2nd edition, 2018.