Chapter 5 -2

เรียนรู้เทคนิคการจำแนกประเภทข้อมูล (Data Classification)

Introduction to Datamining and Warehousing

Asst.Prof.Wilairat Yathongchai

Scope

- Decision Tree Induction
- เทคนิคการจำแนกประเภทข้อมูลด้วย Naïve Bays
- เทคนิคการจำแนกประเภทข้อมูลด้วย K-Nearest Neighbor
- ตัวอย่างงานและการวิเคราะห์ผลที่ได้จาก K-Nearest Neighbor, Naïve Bays

Decision Tree Induction

- **Decision Tree Induction คือ** กระบวนการสร้าง **Decision Tree** ซึ่งเป็นโมเดลที่ใช้สำหรับการจำแนกประเภท (Classification) หรือการทำนายค่า (Regression) โดยอาศัยโครงสร้างต้นไม้ (Tree Structure) ในการตัดสินใจ
- **ขั้นตอนการสร้าง Decision Tree จาก Training Datasets เพื่อใช้จำแนกข้อมูล มีดังนี้**
  - 1. เลือก Attribute ที่ทำหน้าที่เป็น Root Node
  - 2. จาก Root Node สร้างเส้นเชื่อมโยงไปยังโหนดลูก จำนวนเส้นเชื่อมโยง จะเท่ากับจำนวนค่าที่เป็นไปได้ทั้งหมดของ Attribute ที่เป็น Root Node
  - 3. ถ้าโหนดลูกเป็นกลุ่มของข้อมูลที่อยู่ในคลาสเดียวกันทั้งหมด ให้หยุดสร้างต้นไม้ แต่ถ้าโหนดลูกมีข้อมูลของหลายคลาสปะปนกันอยู่ ต้องสร้าง Subtree เพื่อจำแนกข้อมูลต่อไป โดยเลือก Subtree มาทำหน้าที่เป็น Root node ของ Subtree มาทำซ้ำในขั้นตอนที่ 2,3

Decision Tree Induction (2)

- **โครงสร้างของ Decision Tree**
- **Node (โหนด)**:
  - **Root Node**: โหนดเริ่มต้นของต้นไม้
  - **Internal Node**: โหนดที่ทำหน้าที่เป็นจุดตัดสินใจ (Decision Point) โดยใช้คุณสมบัติ (Feature) และเงื่อนไข (Condition)
  - **Leaf Node**: โหนดปลายสุดที่แสดงผลลัพธ์ เช่น คลาสที่คาดการณ์ หรือค่าที่ทำนาย
- **Branch (กิ่ง)**: เส้นทางที่เชื่อมระหว่างโหนด ซึ่งแสดงถึงผลลัพธ์ของเงื่อนไขในโหนดก่อนหน้า

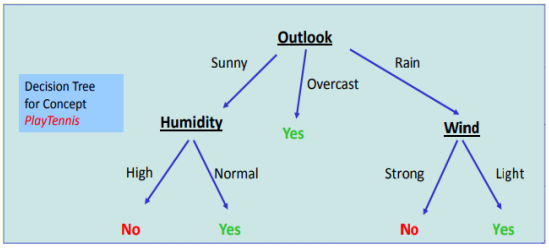
Which Attribute to SELECT?

- ข้อมูลที่กำหนดในตารางเป็นข้อมูลสภาพอากาศที่ใช้ประกอบการตัดสินใจในการเล่นกีฬาชนิดหนึ่งว่า
- มีสภาพอากาศอย่างไรจึงจะเล่น (play = yes)
- มีสภาพอากาศอย่างไรจึงไม่เล่น (play = no)
- ข้อมูลที่เป็นจุดมุ่งหมายในการจำแนก (Class) คือ play
- โดยแอทริบิวท์outlook temperature humidity windy ทำหน้าที่เป็น Predicting Attributes
- ปัญหาที่ต้องพิจารณาคือ จะเลือกAttributes ใดทำหน้าที่เป็น Root Node ในแต่ขั้นตอนของการสร้าง Tree และ Subtree

outlook	temperature	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Which Attribute to SELECT? (2)

- **Attribute Selection**
- **Outlook?**
- **Temperature?**
- **Humidity?**
- **Windy?**



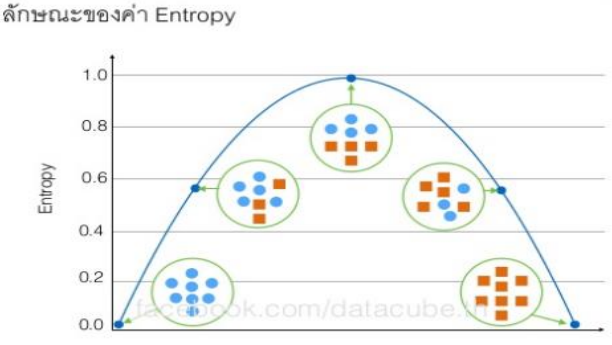
outlook	temperature	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Which Attribute to SELECT? (3)

- **Attribute Selection**
- การสร้างโมเดล Decision Tree จะทำการคัดเลือกแอทริบิวท์ที่มีความสัมพันธ์กับคลาสมากที่สุดขึ้นมาเป็นโหนดบนสุดของTree (Root Node) หลังจากนั้นก็จะหาแอตทริบิวต์ถัดไปเรื่อยๆ ในการหาความสัมพันธ์ของแอทริบิวท์นี้จะใช้ตัววัดที่เรียกว่า Information Gain (IG) ซึ่งถูกนำมาใช้ในการเลือกแอทริบิวท์ในแต่ละ Node ของ Tree โดยแอทริบิวท์ตัวใดที่มีค่าInformation Gain สูงสุดจะถูกเลือก ค่านี้คำนวณได้จากสมการดังนี้
- IG (parent, child) = entropy(parent) – [p(c1) × entropy(c1) + p(c2) × entropy(c2) + …]
- โดยที่ entropy(c1) = -p(c1) log p(c1)
  และ p(c1) คือ ค่าความน่าจะเป็นของc1

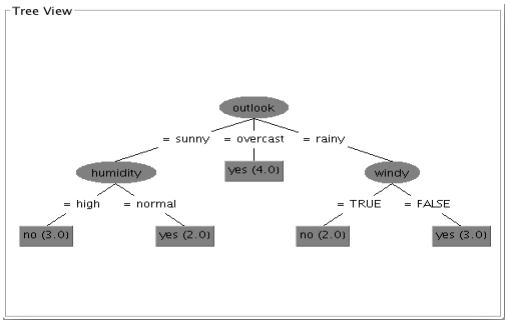
Which Attribute to SELECT? (4)

- จากรูปแต่ละจุดคือข้อมูลแต่ละตัว จะเห็นว่าถ้าข้อมูลมีคำตอบหรือคลาสเดียวกัน เช่น เป็นคลาสสีฟ้า หรือ สีส้มทั้งหมดจะมีค่า Entropy ที่ต่ำที่สุด คือ Entropy เท่ากับ 0 แต่ถ้ามีความแตกต่างกันมาก เช่น เป็นคลาสสีฟ้าครึ่งหนึ่งและคลาสสีส้มอีกครึ่งหนึ่งจะมีค่า Entropy สูงสุด คือ Entropy เท่ากับ 1



Which Attribute to SELECT? (5)

- ทั้งหมดนี้คือขั้นตอนการสร้างโมเดล decision tree ซึ่งข้อดีของโมเดลนี้มีดังนี้
- เป็นโมเดลที่เข้าใจง่าย สามารถแปลความจากโมเดลได้เลย เช่น ถ้าวันไหนที่สภาพอากาศเป็นแบบ outlook แล้วจะมีการจัดแข่งขันกีฬา
- โมเดลที่สร้างได้คัดเลือกแอตทริบิวต์ที่มีความสัมพันธ์กับคลาสคำตอบมาแล้ว ดังนั้นอาจจะไม่ได้ใช้ทุกแอตทริบิวต์ในข้อมูล training



หลักการทำงานของ Decision Tree Induction

1. **การเลือกคุณสมบัติ (Feature Selection)**:
   เลือกคุณสมบัติที่เหมาะสมที่สุดในการแบ่งข้อมูลในแต่ละโหนด โดยใช้เกณฑ์ต่าง ๆ เช่น:
   1. **Information Gain** (ใช้ในอัลกอริทึม ID3)
   2. **Gini Index** (ใช้ใน CART)
   3. **Gain Ratio** (ใช้ใน C4.5)
2. **การแบ่งข้อมูล (Splitting)**:
   แบ่งข้อมูลออกเป็นกลุ่มย่อยตามเงื่อนไขที่กำหนดในโหนด
3. **การหยุดการแบ่ง (Stopping Criteria)**:
   หยุดการแบ่งข้อมูลเมื่อ:
   1. ข้อมูลในโหนดมีคลาสเดียวกันทั้งหมด
   2. ไม่มีคุณสมบัติที่เหมาะสมสำหรับการแบ่ง
   3. ถึงความลึกสูงสุดของต้นไม้ (Maximum Depth)
4. **การตัดแต่งต้นไม้ (Pruning)**:
   ลดความซับซ้อนของต้นไม้เพื่อลดปัญหา Overfitting โดยการลบโหนดที่ไม่จำเป็นออก

Classification Techniques

- **Naïve Bayes**
- เป็นการเรียนรู้แบบ Supervised Learning
- ใช้ในการวิเคราะห์หาความน่าจะเป็นของสิ่งที่ยังไม่เคยเกิดขึ้นโดยการคาดเดาจากสิ่งที่เกิดขึ้นมาก่อน
- โดยอัลกอริทึมจะเรียนรู้จาก Training Set นำสิ่งที่เรียนรู้นั้นมาทำนายในสิ่งที่อยากจะรู้คือ ทำนาย A
- เพื่อทำนายว่า Test Data Instance มีความน่าจะเป็นในการเป็นคลาสแต่ละคลาสเท่าไร
- การเรียนรู้แบบเบย์และต้นไม้ตัดสินใจเป็นเทคนิคที่นิยมใช้ในการวิเคราะห์พยากรณ์ และจำแนกลักษณะข้อมูล

Classification Techniques

- **Naïve Bayes**
- อัลกอริทึมนาอีฟเบย์ หมายถึง เครื่องจักรเรียนรู้ที่อาศัยหลักการความน่าจะเป็น ตามทฤษฎีของเบย์ (Bayes Theorem) ซึ่งมีอัลกอริทึมที่ไม่ซับซ้อน เป็นขั้นตอนวิธีในการจำแนกข้อมูลโดยการเรียนรู้ปัญหาที่เกิดขึ้น เพื่อนำมาสร้างเงื่อนไขการจำแนกข้อมูลใหม่ หลักการของนาอีฟเบย์ใช้การคำนวณหาความน่าจะเป็นในการทำนายผลเป็นเทคนิคในการแก้ปัญหาแบบจำแนกประเภทที่สามารถคาดการณ์ผลลัพธ์ได้ จะทำการวิเคราะห์ความสัมพันธ์ระหว่างตัวแปรเพื่อใช้ในการสร้างเงื่อนไขความน่าจะเป็นสำหรับแต่ละความสัมพันธ์เหมาะกับกรณีของเซตตัวอย่างที่มีจำนวนมาก และคุณสมบัติ (Attribute) ของตัวอย่างไม่ขึ้นต่อกัน โดยกำหนดให้ความน่าจะเป็นของข้อมูลเท่ากับ สมการ

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Classification Techniques (2)

Naïve Bayes

- ความน่าจะเป็น (probability)
  - โอกาสที่เกิดเหตุการณ์จากเหตุการณ์ทั้งหมด ใช้สัญลักษณ์ P() หรือ Pr()
  - โยนเหรียญบาท (มีหัวและก้อย)
    - โอกาสได้หัว มีค่าความน่าจะเป็น 1/2 = 0.5
    - โอกาสได้ก้อย มีค่าความน่าจะเป็น 1/2 = 0.5
  - ความน่าจะเป็นของการพบ spam email
    - มี email ทั้งหมด 100 ฉบับ
    - มี spam email ทั้งหมด 20 ฉบับ
    - มี normal email ทั้งหมด 80 ฉบับ
    - โอกาสที่ emai จะเป็น spam มีความน่าจะเป็น 20/100 = 0.2 หรือ P(spam) = 0.2
    - โอกาสที่ emai จะเป็น normal มีความน่าจะเป็น 80/100 = 0.8 หรือ P(normal) = 0.8

Classification Techniques (3)

Naïve Bayes

- เล่น / ไม่เล่น Tennis

outlook	temperature	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	mild	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Classification Techniques (4)

- **Naïve Bayes**

outlook	temperature	humidity	windy	play
sunny	hot	high	FALSE	?

- **คำถาม** ต้องการรู้ว่าถ้า แอททริบิวต์ outlook = sunny
- แอททริบิวต์ temperature = hot
- แอททริบิวต์ humidity = high
- แอททริบิวต์ windy = FALSE

- มีความน่าจะเป็นที่จะเล่น / ไม่เล่น Tennis ?

Classification Techniques (5)

- **Naïve Bayes : ขั้นตอนการคำนวณ**

1. คำนวณหาความน่าจะเป็นในการเล่น / ไม่เล่น Tennis ตามทัศนวิสัยทั้ง 14 วัน
2. คำนวณหาความน่าจะเป็นในการเล่น / ไม่เล่น Tennis ของคำถาม
   - แอททริบิวต์ outlook = sunny
   - แอททริบิวต์ temperature = hot
   - แอททริบิวต์ humidity = high
   - แอททริบิวต์ windy = FALSE

Classification Techniques (6)

- **Naïve Bayes : ขั้นตอนการคำนวณ**

P(**play** = yes) = 9/14 = 0.64
P(**play** = no) = 5/14 = 0.36

attribute	**play** = yes	**play** = no
outlook = sunny	2/9 = 0.22	3/5 = 0.60
outlook = overcast	4/9 = 0.45	0/5 = 0.00
outlook = rainy	3/9 = 0.33	2/5 = 0.40
temperature = hot	2/9 = 0.22	2/5 = 0.40
temperature = mild	4/9 = 0.45	2/5 = 0.40
temperature = cool	3/9 = 0.33	1/5 = 0.20
humidity = high	3/9 = 0.33	4/5 = 0.80
humidity = normal	6/9 = 0.67	1/5 = 0.20
windy = TRUE	3/9 = 0.33	3/5 = 0.60
windy = FALSE	6/9 = 0.67	2/5 = 0.40

Classification Techniques (7)

- **Naïve Bayes : Prediction on unseen data**
- **ต้องคำนวณค่าความน่าจะเป็นที่มีแอตทริบิวต์เหล่านี้แล้วตอบคลาส play = yes**

P(**play** = yes|A) = P(**outlook** = sunny|play = yes) x P(**temperature** = hot|play = yes) x
P(**humidity** = high|play = yes) x P(**windy** = FALSE|play = yes) x
P(play = yes)
= 0.22 x 0.22 x 0.33 x 0.67 x 0.64
= 0.0068

- **ต้องคำนวณค่าความน่าจะเป็นที่มีแอตทริบิวต์เหล่านี้แล้วตอบคลาส play = No**

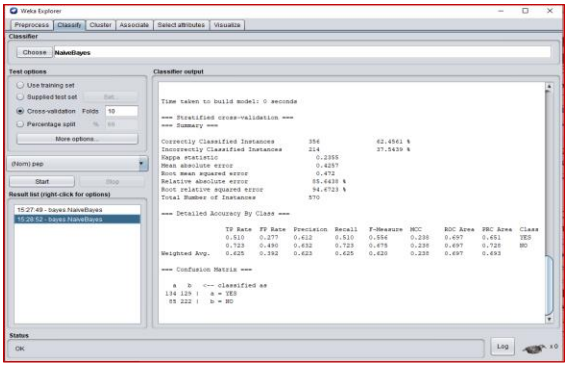
P(**play** = no|A) = P(**outlook** = sunny|play = no) x P(**temperature** = hot|play = no) x
P(**humidity** = high|play = no) x P(**windy** = FALSE|play = no) x
P(play = no)
= 0.60 x 0.40 x 0.80 x 0.40 x 0.36
= 0.0276

- **เมื่อเปรียบเทียบค่าความน่าจะเป็นที่ได้จาก 2 คลาสแล้วพบว่าค่า P(play = no|A) (=0.0276) มีค่ามากกว่า P(play = yes|A) (=0.0068) ดังนั้นโมเดลของเราจึงทำนายว่าข้อมูล instance นี้มีค่าคลาส play = no**

Classification Techniques (8)

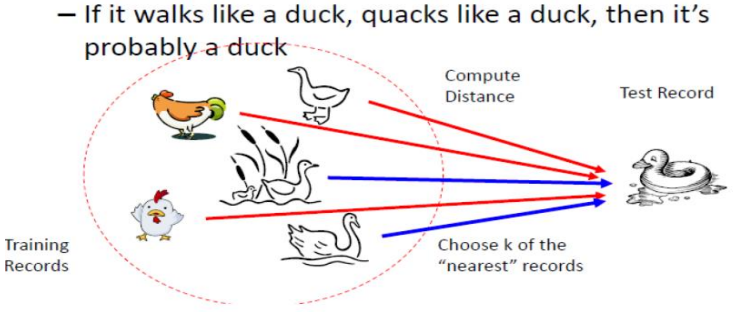
- **Naïve Bayes in Weka**
- โหลดไฟล์ bank-data.arff (ใน AssignmentII)
- เลือก tab classify
- คลิก weka→classifiers →bayes → NaiveBayes
- เลือกวิธีการแบ่งข้อมูล (10-fold cross validation)
- คลิก start
- อธิบายผลที่ได้

Criteria	C4.5	Naïve Bayes
Time to build the model (seconds)		
Correctly classify		
Incorrectly classify		
Accuracy		
Precision for yes		
Precision for no		



Classification Techniques

- **K-Nearest Neighbor (KNN)**
- ใช้หลักการเปรียบเทียบข้อมูลที่สนใจกับข้อมูลอื่นว่ามีความคล้ายคลึงมากน้อยเพียงใด หากข้อมูลที่กำลังสนใจ อยู่ใกล้ข้อมูลใดมากที่สุด ระบบจะให้คำตอบเป็นเหมือนคำตอบของข้อมูลที่อยู่ใกล้ที่สุดนั้น เป็นวิธีการที่ไม่ซับซ้อนและเข้าใจง่ายที่สุดที่ใช้ในการจำแนกประเภทข้อมูล



Classification Techniques

- การหาเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbors) หมายถึง วิธีที่ใช้ในการจัดแบ่งคลาส โดยเทคนิคนี้จะตัดสินใจว่าคลาสใดที่จะแทนเงื่อนไขหรือกรณีใหม่ ๆ ได้ โดยการตรวจสอบจำนวนบางจำนวน “K” ในขั้นตอนวิธีการหาเพื่อนบ้านใกล้สุด ของกรณีหรือเงื่อนไขที่เหมือนกันหรือใกล้เคียงกันมากที่สุด โดยจะหาผลรวม (Count Up) ของจำนวนเงื่อนไข หรือกรณีต่าง ๆ สำหรับแต่ละคลาส และกำหนดเงื่อนไขใหม่ให้คลาสที่เหมือนกันกับคลาสที่ใกล้เคียงกันมากที่สุด การนำเทคนิคของขั้นตอน KNNไปใช้นั้นเป็นการหาระยะห่างระหว่างแต่ละตัวแปร (Attribute) ซึ่งวิธีนี้เหมาะสำหรับข้อมูลแบบตัวเลขแต่ตัวแปรที่เป็นค่าแบบไม่ต่อเนื่องก็สามารถทำได้เพียง

Classification Techniques (2)

- **K-Nearest Neighbor (KNN)** การนำเทคนิคของ K-NN ไปใช้เป็นการหาวิธีการวัดระยะห่างระหว่างแต่ละAttributeในข้อมูลให้ได้ และจากนั้นคำนวณค่าออกมา
- เป็นการเรียนรู้โดยพิจารณาจากตัวอย่าง (Instance-based Learning)
- ข้อมูลฝึกถูกนำมาใช้ในการจำแนกข้อมูลใหม่โดยเปรียบเทียบจากลักษณะความคล้ายคลึงกันของข้อมูล จากค่าน้ำหนักโดยการพิจาณาระยะห่างระหว่างข้อมูลที่สนใจกับข้อมูลที่อยู่ใกล้สุด k ตัว ร่วมด้วย
- ถูกเรียกว่า “Lazy learning”
- เป็นวิธีที่ง่ายและมีประสิทธิภาพ แต่การประมวลผลช้า

Classification Techniques (3)

- **K-Nearest Neighbor (KNN)**

- การใช้โมเดลเพื่อ predict ข้อมูลใหม่

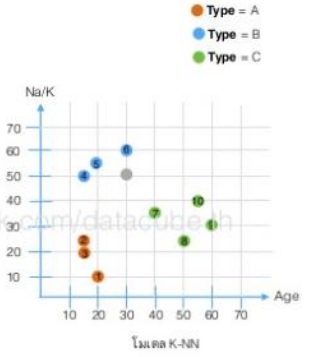
ID	Na/K	Age
11	50	30

unseen data

ID	Na/K	Age	Type
1	10	20	A
2	25	15	A
3	20	15	A
4	50	15	B
5	55	20	B
6	60	30	B
7	35	40	C
8	25	50	C
9	30	60	C
10	40	55	C

training data

ID	ระยะห่างจากข้อมูล ID = 11
1	$\sqrt{(10-50)^2 + (20-30)^2} = 41.23$
2	$\sqrt{(25-50)^2 + (15-30)^2} = 29.15$
3	$\sqrt{(20-50)^2 + (15-30)^2} = 33.54$
4	$\sqrt{(50-50)^2 + (15-30)^2} = 15.00$
5	$\sqrt{(55-50)^2 + (20-30)^2} = 11.18$
6	$\sqrt{(60-50)^2 + (30-30)^2} = 10.00$
7	$\sqrt{(35-50)^2 + (40-30)^2} = 18.03$
8	$\sqrt{(25-50)^2 + (50-30)^2} = 32.02$
9	$\sqrt{(30-50)^2 + (60-30)^2} = 36.06$
10	$\sqrt{(40-50)^2 + (55-30)^2} = 26.93$



Classification Techniques (4)

- **K-Nearest Neighbor (KNN)**

- การใช้โมเดลเพื่อ predict ข้อมูลใหม่ โดยกำหนด K = 1

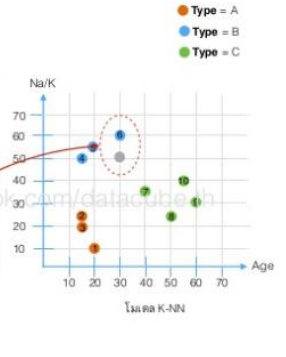
ID	Na/K	Age
11	50	30

unseen data

ID	Na/K	Age	Type
1	10	20	A
2	25	15	A
3	20	15	A
4	50	15	B
5	55	20	B
6	60	30	B
7	35	40	C
8	25	50	C
9	30	60	C
10	40	55	C

training data

ID	ระยะห่างจากข้อมูล ID = 11
1	$\sqrt{(10-50)^2 + (20-30)^2} = 41.23$
2	$\sqrt{(25-50)^2 + (15-30)^2} = 29.15$
3	$\sqrt{(20-50)^2 + (15-30)^2} = 33.54$
4	$\sqrt{(50-50)^2 + (15-30)^2} = 15.00$
5	$\sqrt{(55-50)^2 + (20-30)^2} = 11.18$
6	$\sqrt{(60-50)^2 + (30-30)^2} = 10.00$
7	$\sqrt{(35-50)^2 + (40-30)^2} = 18.03$
8	$\sqrt{(25-50)^2 + (50-30)^2} = 32.02$
9	$\sqrt{(30-50)^2 + (60-30)^2} = 36.06$
10	$\sqrt{(40-50)^2 + (55-30)^2} = 26.93$



Classification Techniques (5)

- **K-Nearest Neighbor (KNN) : Example**

จากการสุ่มเลือกข้อมูล Iris Dataset ซึ่งมีข้อมูลทั้งสิ้น 150 รายการ แบ่งออกเป็น 3 คลาส คลาสละ 50 รายการ เพื่อมาใช้เป็นTraining Dataset จำนวน 9 รายการ โดยทำการสุ่มเลือกมาจากแต่ละคลาส ๆ ละ 3 รายการ ข้อมูลหลังจากการสุ่มเลือกแล้ว ดังตารางต่อไปนี้

Sepal Length	Sepal Width	Petal Length	Petal Width	Class
48	30	14	1	Iris-setosa
51	35	14	3	Iris-setosa
50	34	16	4	Iris-setosa
66	30	44	14	Iris-versicolor
67	31	47	15	Iris-versicolor
58	26	40	12	Iris-versicolor
77	26	69	23	Iris-virginica
77	30	61	23	Iris-virginica
67	30	52	23	Iris-virginica

Classification Techniques (6)

- **K-Nearest Neighbor (KNN) : Example**

จงแสดงวิธีการคำนวณหา Class ของข้อมูลในตารางข้างล่าง โดยใช้รูปแบบ K-Nearest Neighbors(KNN) (กำหนดให้ K=3)

Sepal Length	Sepal Width	Petal Length	Petal Width	Class (KNN)
56	37	13	4	?

Solution

นำข้อมูลที่ต้องการทำนายผลมาเปรียบเทียบกับข้อมูลใน Training Dataset โดยคำนวณหาค่า Euclidean Distance ตามสูตร

$$\text{distance} = \sqrt{\sum_{k=1}^{n}(p_k - q_k)^2}$$

Classification Techniques (7)

- **K-Nearest Neighbor (KNN) : Example**

ผลลัพธ์ที่ได้

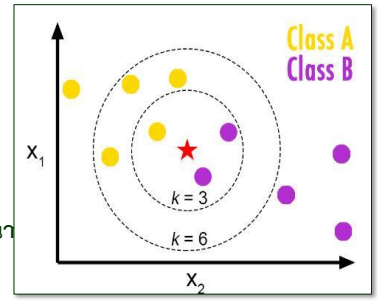
$$\text{distance}(x,R_1) = \sqrt{(56-48)^2 + (37-30)^2 + (13-14)^2 + (4-1)^2} = 11.09$$
$$\text{distance}(x,R_2) = \sqrt{(56-51)^2 + (37-35)^2 + (13-14)^2 + (4-3)^2} = 5.57$$
$$\text{distance}(x,R_3) = \sqrt{(56-50)^2 + (37-34)^2 + (13-16)^2 + (4-4)^2} = 7.35$$
$$\text{distance}(x,R_4) = \sqrt{(56-66)^2 + (37-30)^2 + (13-44)^2 + (4-14)^2} = 34.79$$
$$\text{distance}(x,R_5) = \sqrt{(56-67)^2 + (37-31)^2 + (13-47)^2 + (4-15)^2} = 37.87$$
$$\text{distance}(x,R_6) = \sqrt{(56-58)^2 + (37-26)^2 + (13-40)^2 + (4-12)^2} = 30.30$$
$$\text{distance}(x,R_7) = \sqrt{(56-77)^2 + (37-26)^2 + (13-69)^2 + (4-23)^2} = 63.71$$
$$\text{distance}(x,R_8) = \sqrt{(56-77)^2 + (37-30)^2 + (13-61)^2 + (4-23)^2} = 56.17$$
$$\text{distance}(x,R_9) = \sqrt{(56-67)^2 + (37-30)^2 + (13-52)^2 + (4-23)^2} = 45.30$$

จากการคำนวณพบว่า ข้อมูลมีความใกล้เคียงกับ Training dataset ในเรคอร์ดที่ 1,2 และ 3 ซึ่งมี class = Iris Setosa ดังนั้นจึงทำนายได้ว่าข้อมูลที่ใช้ในการทดสอบจากโจทย์ จะมี class = Iris Setosa

Classification Techniques (8)

- **K-Nearest Neighbor (KNN) : Example**
- การจำแนกข้อมูลที่เลือกเฉพาะข้อมูลที่มีระยะห่าง 1 กลุ่ม (ใกล้ที่สุด) จะเรียกว่า “1NN (One Nearest Neighbor) ดังนั้น “k–NN” ค่า k จึงเป็นจำนวนของกลุ่มที่ต้องการเลือกเป็นกลุ่มเพื่อนบ้าน โดยควรกำหนดเป็นเลขคี่ สำหรับการหาค่าระยะทางจะใช้สมการจากทฤษฎีการวัดระยะทางของ Euclidean ดังนี้

$$d = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$$



- เมื่อ p คือค่าของชุดข้อมูลที่ต้องการจำแนก
  q คือค่าของชุดข้อมูลเพื่อนบ้านที่นำมาพิจารณา

Classification Techniques (9)

- **K-Nearest Neighbor (KNN) : ขั้นตอนวิธี**
- ข้อมูลใหม่ (Unknown) ซึ่งไม่ทราบ Class เรียกว่า U
- ข้อมูลชุดสอน (Training set) มีขนาดเท่ากับ Nrow * Marrtibue
- วนรอบ จำนวน N รอบ
  - คำนวณหาระยะห่างของ U กับ Training[i]
- จบการทำงาน
- คำนวณหาระยะทางที่ใกล้ที่สุด จำนวน k ค่า
- เลือกคำตอบจากชุดข้อมูลสอนที่ใกล้ที่สุด หรือมีคำตอบซ้ำกันมากที่สุด

Classification Techniques (10)

- **K-Nearest Neighbor (KNN) : ตัวอย่าง**

Attributes x								Attribute y
ชื่อ	อุณหภูมิร่างกาย	ผิวหนัง	การเกิดเป็นตัว	อาศัยในน้ำ	บิน	มีขาหรือไม่	จำศีล	คลาส
มนุษย์	เลือดอุ่น	มีขน	ใช่	ไม่	ไม่	ใช่	ไม่	Mammal
งูเหลือม	เลือดเย็น	เกล็ด	ไม่	ไม่	ไม่	ไม่	ใช่	Reptile
แซลมอน	เลือดเย็น	เกล็ด	ไม่	ใช่	ไม่	ไม่	ไม่	Fish

Data transformation

Attributes x								Attribute y
ชื่อ	อุณหภูมิร่างกาย (อุ่น)	ผิวหนัง (มีขน)	การเกิดเป็นตัว	อาศัยในน้ำ (ใช่)	บิน (ใช่)	มีขาหรือไม่ (ใช่)	จำศีล (ใช่)	คลาส
มนุษย์	1	1	1			1		Mammal
งูเหลือม							1	Reptile
แซลมอน				1				Fish

Classification Techniques (11)

- **K-Nearest Neighbor (KNN) : ตัวอย่าง (1NN)**

	Attributes x							Attribute y
ชื่อ	อุณหภูมิร่างกาย (อุ่น)	ผิวหนัง (มีขน)	การเกิดเป็นตัว	อาศัยในน้ำ (ใช่)	บิน (ใช่)	มีขาหรือไม่ (ใช่)	จำศีล (ใช่)	คลาส
มนุษย์	1	1	1			1		Mammal
งูเหลือม							1	Reptile
แซลมอน				1				Fish

ชื่อ	อุณหภูมิร่างกาย (อุ่น)	ผิวหนัง (มีขน)	การเกิดเป็นตัว	อาศัยในน้ำ (ใช่)	บิน (ใช่)	มีขาหรือไม่ (ใช่)	จำศีล (ใช่)	คลาส
สัตว์ประหลาด	1		1					???

Classification Techniques (12)

- **K-Nearest Neighbor (KNN) : ตัวอย่าง (1NN)**

$$d = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

ระยะทางของชุดข้อมูลมนุษย์

$$d_1 = \sqrt{(1-1)^2 + (0-1)^2 + (1-1)^2 + (0-0)^2 + (0-0)^2 + (0-1)^2 + (0-0)^2} = 1.414214$$

ระยะทางของชุดข้อมูลงูเหลือม

$$d_2 = \sqrt{(1-0)^2 + (0-0)^2 + (1-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-1)^2} = 1.732051$$

ระยะทางของชุดข้อมูลแซลมอน

$$d_3 = \sqrt{(1-0)^2 + (0-0)^2 + (1-0)^2 + (0-1)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} = 1.732051$$

Classification Techniques (13)

- **K-Nearest Neighbor (KNN) : ตัวอย่าง (3NN)**

	Attributes x							Attribute y
ชื่อ	อุณหภูมิร่างกาย (อุ่น)	ผิวหนัง (มีขน)	การเกิดเป็นตัว	อาศัยในน้ำ (ใช่)	บิน (ใช่)	มีขาหรือไม่ (ใช่)	จำศีล (ใช่)	คลาส
มนุษย์	1	1	1			1		Mammal
งูเหลือม							1	Reptile
แซลมอน				1				Fish
วาฬ	1	1	1	1				Mammal
ค้างคาว	1	1	1		1			Mammal
ปลาไหล				1				Fish
สัตว์ประหลาด	1		1					???

ชื่อ	ระยะห่างระหว่างสัตว์ประหลาดกับชุดข้อมูล	คลาส
มนุษย์	1.414214	Mammal
งูเหลือม	1.732051	Reptile
แซลมอน	1.732051	Fish
วาฬ	1.414214	Mammal
ค้างคาว	1.414214	Mammal
ปลาไหล	1.732051	Fish

Classification Techniques (14)

- **K-Nearest Neighbor (KNN) : ตัวอย่าง (3NN)**

- ถ้าคำตอบที่ได้ไม่เหมือนกัน สามารถพิจารณาได้โดย
- เลือกคำตอบจากเสียงข้างมาก เช่น สัตว์เลี้ยงลูกด้วยนม สัตว์เลื้อยคลาน สัตว์เลี้ยงลูกด้วยนม จะสรุปว่าเป็นสัตว์เลี้ยงลูกด้วยนมจากเสียงข้างมาก
- เลือกคำตอบจากคำตอบที่มีระยะทางน้อยที่สุด (ในกรณีที่คำตอบไม่เหมือนกันเลย)

Classification Techniques (15)

- **K-Nearest Neighbor (KNN) : เปรียบเทียบการทำงาน**

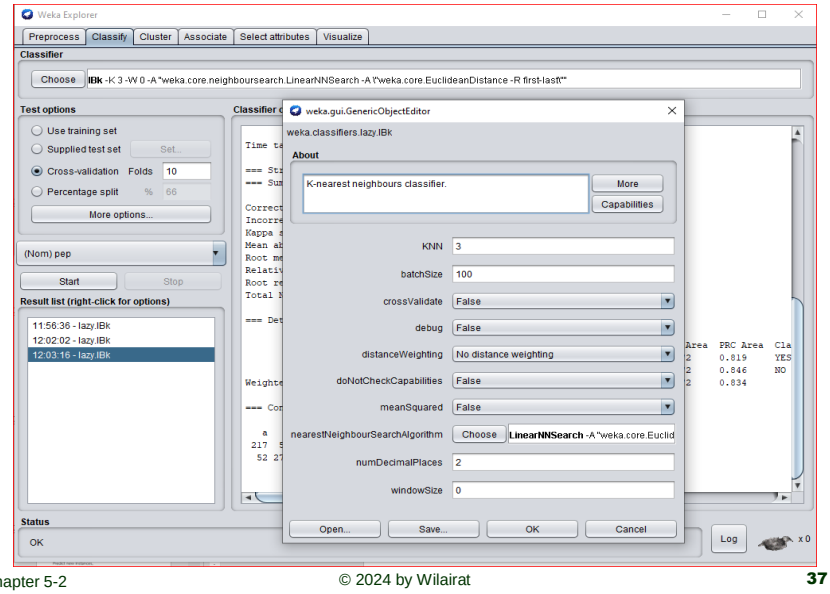
เกณฑ์	D-tree	KNN
ระยะเวลาในการสร้างโมเดล	ใช้เวลานาน	ไม่มีการสร้างโมเดล
ระยะเวลาในการจำแนกข้อมูลใหม่	ใช้เวลารวดเร็ว	ใช้เวลานานเพราะต้องทำการเปรียบเทียบกับชุดข้อมูล
ความยากง่าย	ยากในการสร้างโมเดล	ง่ายในการคำนวณ

Classification Techniques (16)

- **K-Nearest Neighbor (KNN) in Weka**
- โหลดไฟล์ bank.arff
- เลือก Tab classify
- คลิก Weka→classifiers→lazy →Ibk
- คลิก Panel Ibk
- ระบุค่า K ที่ต้องการ กดปุ่ม Ok
- เลือกวิธีการแบ่งข้อมูล
- คลิก start
- อธิบายผลลัพธ์
- เปรียบเทียบผลที่ได้จากทั้ง 3 อัลกอริทึม คือ ID3, Naïve bayes และKNN

Criteria	C4.5	Naïve Bayes
Time to build the model (seconds)		
Correctly classify		
Incorrectly classify		
Accuracy		
Precision		
Recall		

Classification Techniques (17)



Classification Techniques (18)

EXERCISE

1. จงใช้ ข้อมูลตารางต่อไปนี้สำหรับสร้าง Naïve Bayes Classifier

Predict new instances,

1.1 X_1 = (A = 1, B = 1 , C = 1)

1.2 X_2 = (A = 1 , B = 0 , C = 0)

1.3 X_3 = (A = 0, B = 1 , C = 1)

Record	A	B	C	Class
1	0	0	0	+
2	1	0	1	−
3	0	1	1	+
4	1	1	1	−
5	0	0	1	+
6	1	0	1	+
7	1	0	1	−
8	1	1	1	−
9	1	1	1	+
10	1	0	1	+

Classification Techniques (19)

- **K-Nearest Neighbor (KNN) in Weka**

Exercise (K=1,K=3)

Height(cm)	Weight(kg)	Waistline(inch)	Chest(inch)	Gender
165	60	32	37	F
175	75	33	43	M
166	50	30	34	M
155	50	28	32	F
170	60	30	34	?

- **เปรียบเทียบความแตกต่างของ 3 อัลกอริทึม คือ J48 , Naïve Bayes และ KNN (ข้อมูลในรูปไฟล์ Man.arff)**