Phương pháp k láng giềng
K nearest neighbors - KNN

Nội dung

- **Giới thiệu về KNN**
- Kết luận và hướng phát triển

K nearest neighbors – K láng giềng

Phương pháp K láng giềng

Tell me
who your friends are
and I' ll tell you
who you are

K nearest neighbors – K láng giềng

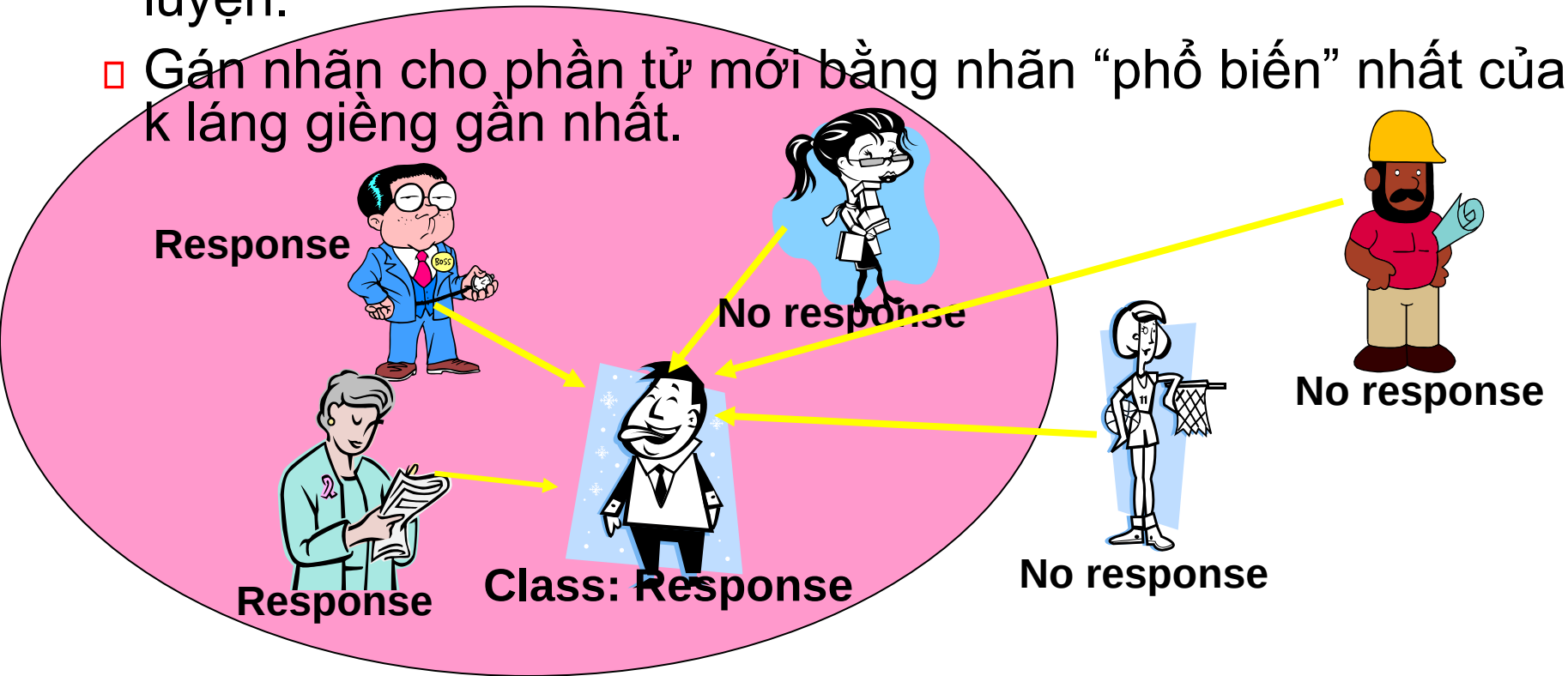
Tell me
who your friends
are and I' ll tell you
who you are

K nearest neighbors – K láng giềng

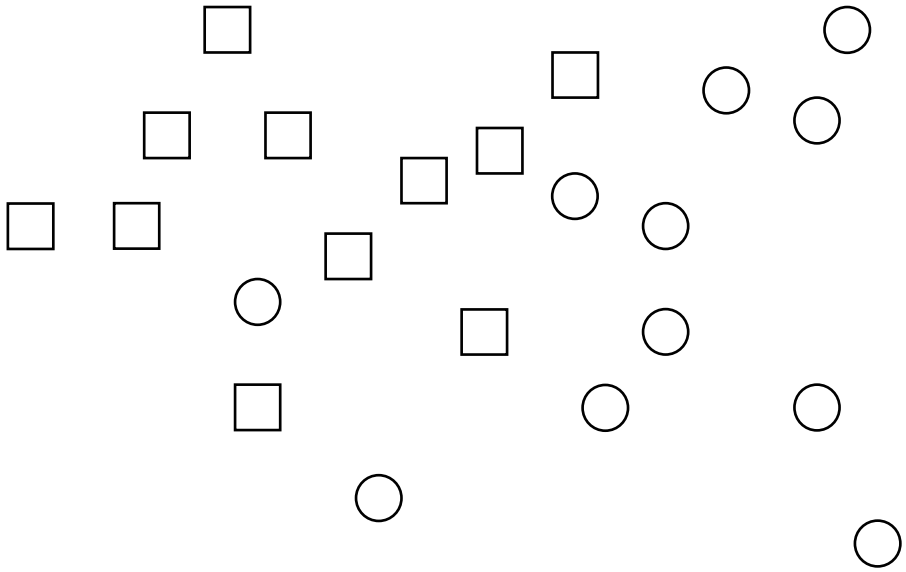
- **K láng giềng - KNN** là một thuật toán phân lớp (classification) các trường hợp mới đến dựa trên một số lượng thông tin "phổ biến" nhất của **k** láng giềng gần nhất với nó.
- K láng giềng (K-Nearest Neighbors) còn được gọi bằng các tên khác như Lazy Learning, Instance-Based Learning
- KNN được Fix và Hodges đề xuất năm 1952
- Học có giám sát (supervised classification)

K-Nearest Neighbor – K láng giềng

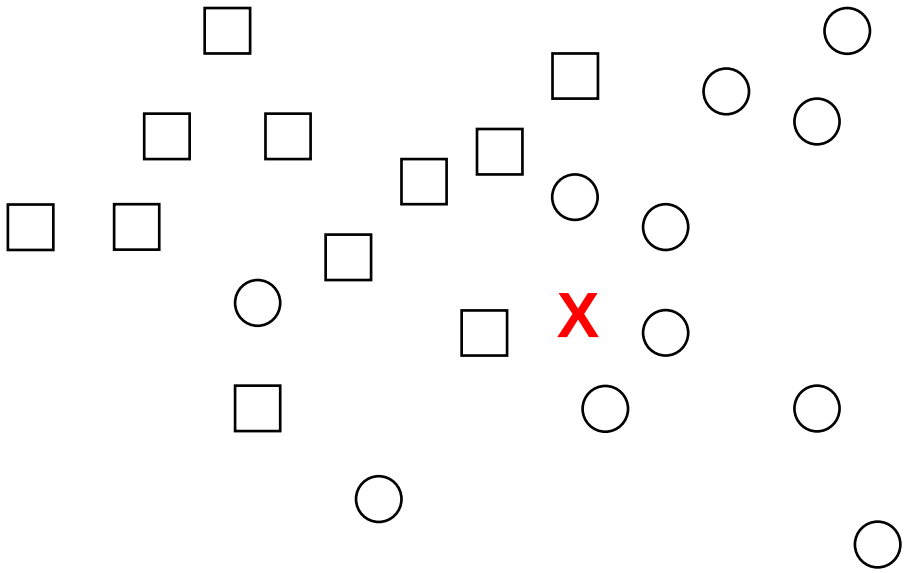
- Để xác định được lớp của phần tử mới đến
  - Tính toán khoảng cách từ phần tử mới đến đến các phần tử còn lại trong tập huấn luyện.
  - Chọn K phần tử gần nhất với phần tử mới trong tập huấn luyện.
  - Gán nhãn cho phần tử mới bằng nhãn “phổ biến” nhất của k láng giềng gần nhất.



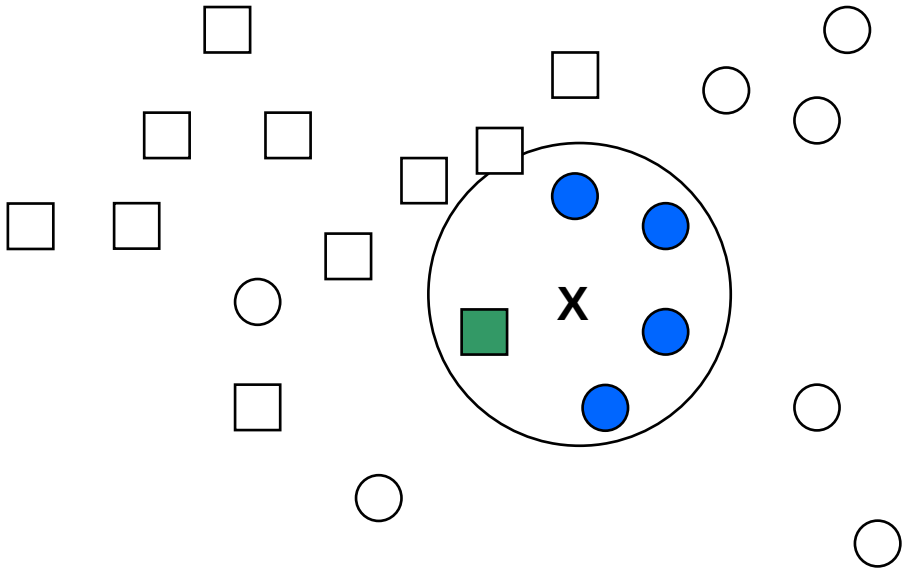
Phương pháp KNN



Phương pháp KNN



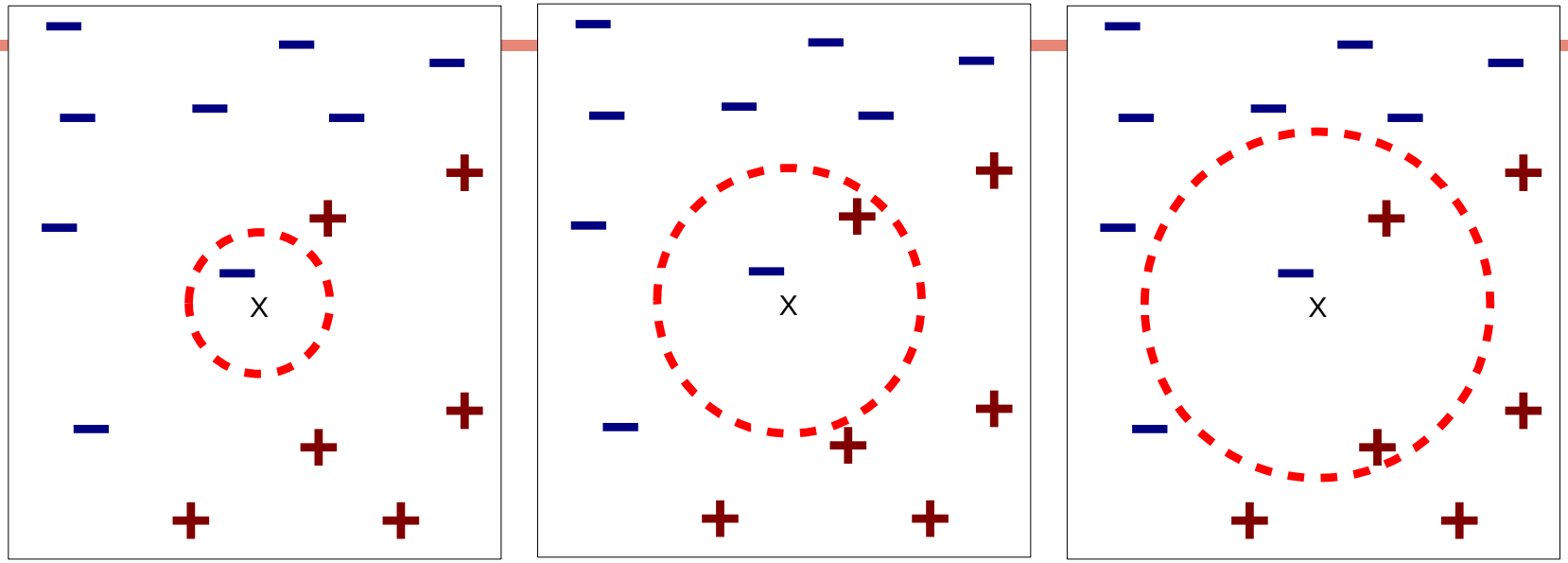
Phương pháp KNN



K =?

Nhãn của phần tử mới đến là hình vuông hay hình tròn?

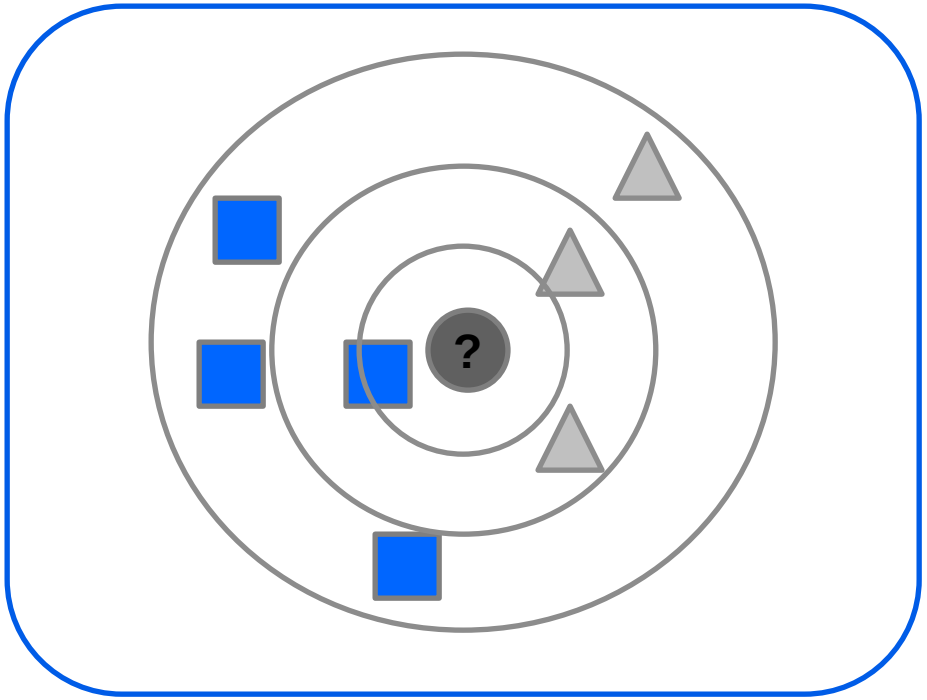
K-Nearest-Neighbor Strategy



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

K-nearest neighbors of a record x are data points that have the k smallest distance to x

Phương pháp KNN



- **Phần tử mới xuất hiện thuộc nhóm nào khi**
- $k = 1$:
- $k = 3$:
- $k = 7$

- **Choosing the value of *k*:**
  - If *k* is too small, sensitive to noise points
  - If *k* is too large, neighborhood may include points from other classes
  - Choose an odd value for *k*, to eliminate ties

Các độ đo khoảng cách

- Khoảng cách được tính theo từng kiểu của dữ liệu
  - Kiểu số,
  - Kiểu rời rạc (nominal type),
  - Nhị phân

Các độ đo khoảng cách - Kiểu số

- Khoảng cách *Minkowski*

$$d(i,j) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + ... + |x_{ip} - x_{jp}|^q)}$$

$i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ và $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ là 2 phần tử dữ liệu trong p-dimensional, q là số nguyên dương

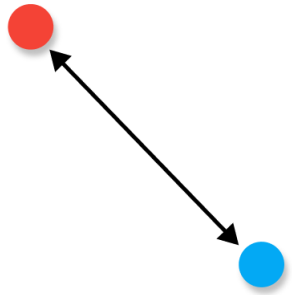
- nếu $q = 1$, d là khoảng cách Manhattan

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + ... + |x_{ip} - x_{jp}|$$

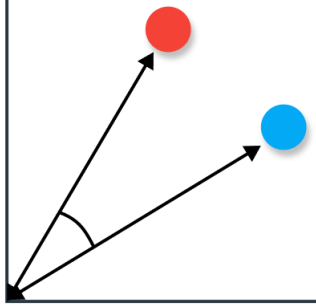
- nếu $q = 2$, d là khoảng cách Euclid

$$d(i,j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + ... + |x_{ip} - x_{jp}|^2)}$$

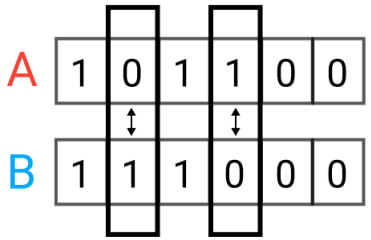
Euclidean



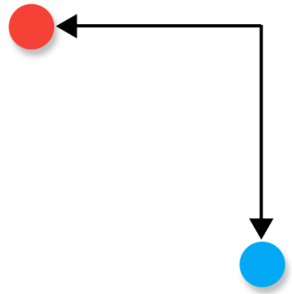
Cosine



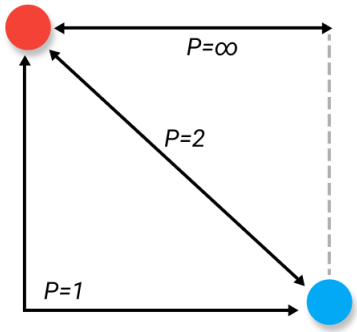
Hamming



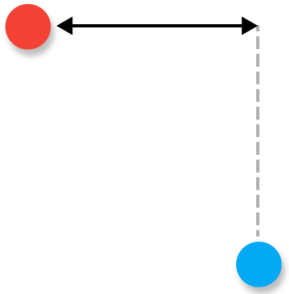
Manhattan



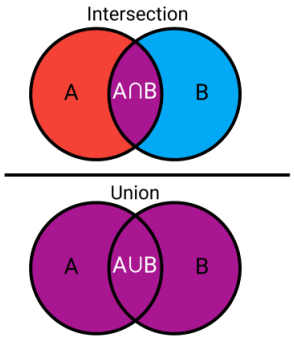
Minkowski



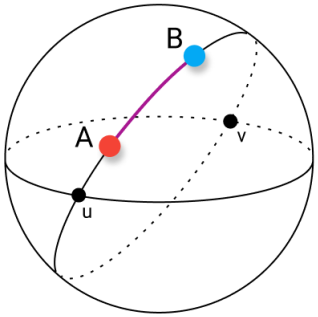
Chebyshev



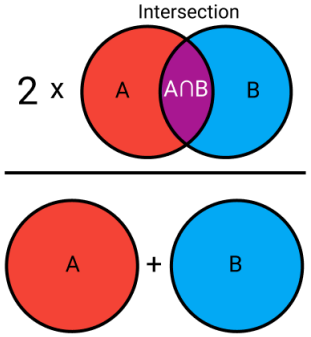
Jaccard



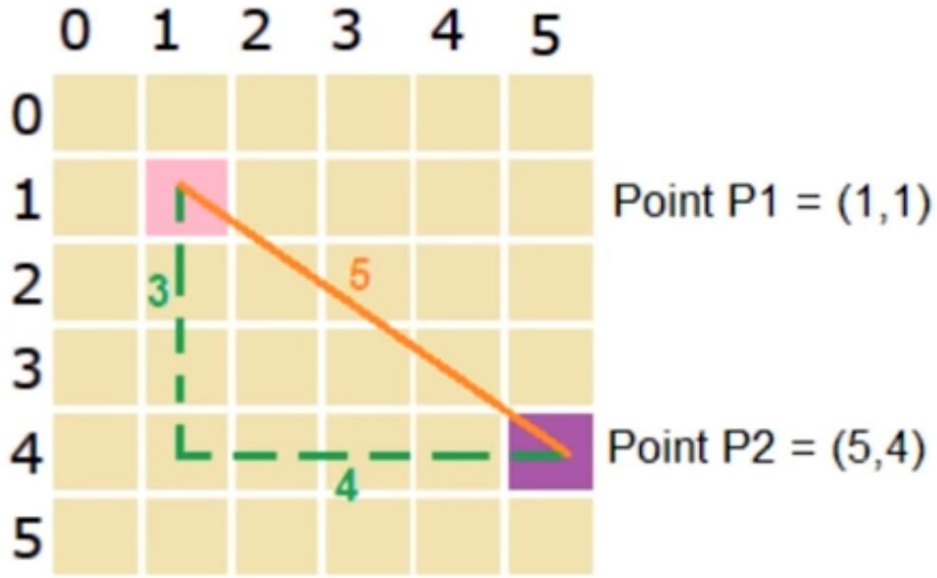
Haversine



Sørensen-Dice



Manhattan Distance



Euclidean distance = $\sqrt{(5-1)^2 + (4-1)^2} = 5$

Manhattan distance = $|5-1| + |4-1| = 7$

Các độ đo khoảng cách –
Kiểu rời rạc (nominal type)

- VD: thuộc tính color có giá trị là red, green, blue, etc.

  - phương pháp matching đơn giản,
    - m là số lượng matches và
    - p là tổng số biến (thuộc tính),
    - khoảng cách được định nghĩa :

$$d(i,j)=\frac{p-m}{p}$$

Kiểu rời rạc (nominal type)

$$d(i,j)=\frac{p-m}{p}$$

- m là số lượng matches và
- p là tổng số biến (thuộc tính),

	Màu tóc	Màu mắt	Chiều cao	Cân nặng	Trình độ
Nam	Đen	Đen	Cao	Trung bình	Cao đẳng
Lan	Nâu	Đen	Thấp	Trung bình	Đại học
Điệp	Nâu	Đen	Cao	Trung bình	Cao đẳng

d(Nam, Lan) =?

d(Nam, Điệp) =?

Proximity Measure for Binary Attributes

- A contingency table for binary data

Object *i* \ Object *j*	1	0	sum
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

- Distance measure for symmetric binary variables:

$$d(i,j) = \frac{r+s}{q+r+s+t}$$

- Distance measure for asymmetric binary variables:

$$d(i,j) = \frac{r+s}{q+r+s}$$

- Jaccard coefficient (***similarity*** measure for *asymmetric* binary variables):

$$sim_{Jaccard}(i,j) = \frac{q}{q+r+s}$$

Dissimilarity between Binary Variables

- **Example**

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- **Let the values Y and P be 1, and the value N 0**

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

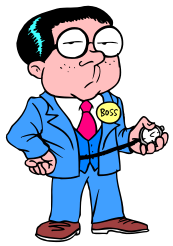
$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

		Jim	
		1	0
Mary	1	1	2
	0	1	2

K nearest neighbors – K láng giềng

Cách tính khoảng cách giữa các phần tử

- **Mỗi phần tử được biểu diễn bởi tập hợp các thuộc tính**



John:
Tuổi = 35
Thu nhập = 95K
Số thẻ tín dụng = 3

Mary:
Tuổi = 41
Thu nhập = 215K
Số thẻ tín dụng = 2

- Sử dụng khoảng cách Euclide giữa 2 phần tử.

$$d(i,j)=\sqrt{(|x_{i1}-x_{j1}|^2+|x_{i2}-x_{j2}|^2+...+|x_{ip}-x_{jp}|^2)}$$

Khoảng cách (John, Mary) =

sqrt [(35-41)^2+(95K-215K)^2 +(3-2)^2]

K nearest neighbors – K láng giềng

- **K = 3 : 3 nearest neighbors**

Customer	Age	Income	No. credit cards	Response
John	35	35K	3	No
Mary	22	50K	2	Yes
Hannah	63	200K	1	No
Tom	59	170K	1	No
Nellie	25	40K	4	Yes
David	37	50K	2	?

K nearest neighbors – K láng giềng

- **K = 3 : 3 nearest neighbors**

Customer	Age	Income (K)	No. cards	Resp	Distance from David
John	35	35	3	No	sqrt [(35-37)²+(35-50)² +(3-2)²]=**15.16**
Mary	22	50	2	Yes	sqrt [(22-37)²+(50-50)² +(2-2)²]=**15**
Hannah	63	200	1	No	sqrt [(63-37)²+(200-50)² +(1-2)²]=**152.23**
Tom	59	170	1	No	sqrt [(59-37)²+(170-50)² +(1-2)²]=**122**
Nellie	25	40	4	Yes	sqrt [(25-37)²+(40-50)² +(4-2)²]=**15.74**
David	37	50	2	? = Yes	

K nearest neighbors – K láng giềng

- **K = 3 : 3 nearest neighbors**

Tên	Tuổi	Thu nhập	Số lượng thẻ tín dụng	Số tiền mua bảo hiểm
John	35	35K	3	6.5tr
Mary	22	50K	2	4.7tr
Hannah	63	200K	1	6.1tr
Tom	59	170K	1	5.4tr
Nellie	25	40K	4	4.5tr
David	37	50K	2	?

K nearest neighbors

- Phương pháp KNN (tên khác instance-based, lazy)
  - rất đơn giản, **không có quá trình học,** Không có mô hình được xây dựng: Lưu trữ tất cả các dữ liệu huấn luyện
  - Hệ thống không thực hiện công việc gì cho đến khi có một trường hợp mới đến cần được phân loại. khi phân loại **mất nhiều thời gian so với một mô hình**, do quá trình tìm kiếm k dữ liệu lân cận, sau đó phân loại dựa trên majority vote (nếu là bài toán **hồi quy, giá trị dự báo tính dựa trên giá trị trung bình k láng giềng**)
  - kết quả phụ thuộc vài việc chọn **khoảng cách** sử dụng

K nearest neighbors

- phương pháp KNN
  - có thể làm việc trên **nhiều loại dữ liệu** khác nhau
  - giải quyết các vấn đề **về phân loại, hồi quy**
  - cho kết quả tốt, tuy nhiên **độ phức tạp** của quá trình phân loại **khá lớn**
  - được ứng dụng thành công trong hầu hết các lãnh vực **tìm kiếm thông tin, nhận dạng, phân tích dữ liệu,** etc.

Phương pháp KNN

X1	X2	Lớp
0.45	5	?

X1	X2	Lớp
0.1	10	+1
0.2	25	+1
0.3	0	+1
0.5	11	-1
0.8	100	-1
0	50	+1
1	70	-1

D(Manhattan)
?
?
?
?
?
?
?

⇨ **Lớp ?**

$$d(i,j)=|x_{i_1}-x_{j_1}|+|x_{i_2}-x_{j_2}|+...+|x_{i_p}-x_{j_p}|$$

Phương pháp KNN

X1	X2	Lớp
0.45	5	?

X1	X2	Lớp	D(Manhattan)
0.1	10	+1	5.35
0.2	25	+1	20.25
0.3	0	+1	**5.15**
0.5	11	-1	6.05
0.8	100	-1	95.35
0	50	+1	45.45
1	70	-1	65.55

1NN ⇨ **lớp = +1**

Nhận xét

- *Thuộc tính X2 có miền giá trị (0..100) trong khi thuộc tính X1 có miền giá trị 0..1*
- *Kết quả phụ thuộc nhiều vào X2*

  (chênh lệch X2 lớn hơn so với X1)
- ***nên chuẩn hóa dữ liệu***

  (chuẩn hóa thuộc tính X2 về giá trị 0..1

new_val = (val – min)/(max – min)

Phương pháp KNN

X1	X2	Lớp
0.45	5	?

X1	X2	Lớp
0.1	0.1	+1
0.2	0.25	+1
0.3	0	+1
0.5	**0.11**	**-1**
0.8	1	-1
0	0.5	+1
1	0.7	-1

1NN

Phương pháp KNN

X1	X2	Lớp
0.45	0.05	?

X1	X2	Lớp	D(Manhattan)
0.1	0.1	+1	0.4
0.2	0.25	+1	0.45
0.3	0	+1	0.2
0.5	**0.11**	**-1**	**0.11**
0.8	1	-1	1.3
0	0.5	+1	0.9
1	0.7	-1	1.2

1NN ⇨ **lớp = -1**

KNN Classification – Distance

Tuổi	Thu nhập	Nhãn	Khoảng cách
25	$40,000	N	
35	$60,000	N	
45	$80,000	N	
20	$20,000	N	
35	$120,000	N	
52	$18,000	N	
23	$95,000	Y	**K=3**
40	$62,000	Y	
60	$100,000	Y	
48	$220,000	Y	
33	$150,000	Y	
48	**$142,000**	**?**	

Euclidean Distance

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

KNN Classification – Distance

Tuổi	Thu Nhập	Nhãn	Khoảng cách
25	$40,000	N	102000
35	$60,000	N	82000
45	$80,000	N	62000
20	$20,000	N	122000
35	$120,000	N	22000
52	$18,000	N	124000
23	$95,000	Y	47000
40	$62,000	Y	80000
60	$100,000	Y	42000
48	$220,000	Y	78000
33	$150,000	Y	8000
48	**$142,000**	**?**	

Euclidean Distance

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Age	Loan	Default	Distance	
25	$40,000	N	102000	
35	$60,000	N	82000	
45	$80,000	N	62000	
20	$20,000	N	122000	
35	$120,000	N	22000	2
52	$18,000	N	124000	
23	$95,000	Y	47000	
40	$62,000	Y	80000	
60	$100,000	Y	42000	3
48	$220,000	Y	78000	
33	$150,000	Y	8000	1
48	**$142,000**	**?**		

Euclidean Distance

$$D = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

Table-1- Euclidean Distance

KNN Classification – Standardized Distance

Tuổi	Thu nhập	Nhãn	Khoảng cách
0.125	0.11	N	
0.375	0.21	N	
0.625	0.31	N	
0	0.01	N	
0.375	0.50	N	
0.8	0.00	N	
0.075	0.38	Y	
0.5	0.22	Y	
1	0.41	Y	
0.7	1.00	Y	
0.325	0.65	Y	
0.7	**0.61**	→ **?**	

Standardized Variable

$$X_s = \frac{X - Min}{Max - Min}$$

KNN Classification – Standardized Distance

Tuổi	Thu nhập	Nhãn	Khoảng cách
0.125	0.11	N	0.7652
0.375	0.21	N	0.5200
0.625	0.31	N	0.3160
0	0.01	N	0.9245
0.375	0.50	N	0.3428
0.8	0.00	N	0.6220
0.075	0.38	Y	0.6669
0.5	0.22	Y	0.4437
1	0.41	Y	0.3650
0.7	1.00	Y	0.3861
0.325	0.65	Y	0.3771
0.7	**0.61**	**?**	

Standardized Variable

$$X_s = \frac{X - Min}{Max - Min}$$

Nội dung

- Giới thiệu về KNN
- **Kết luận và hướng phát triển**

Phương pháp KNN

- thường rất chính xác, nhưng chậm do phải duyệt qua dữ liệu để tìm phần tử gần
- giả sử các thuộc tính có độ quan trọng như nhau
  - gán trọng số quan trọng cho mỗi thuộc tính
- chịu đựng được nhiễu
- thống kê đã sử dụng *k*-NN từ những năm 50s
  - khi dữ liệu lớn ($n \to \infty$) và $k/n \to 0$, lỗi gần với giá trị nhỏ nhất

Hướng phát triển

- tăng tốc cho quá trình tìm k phần tử lân cận
  - cấu trúc index
- chọn thuộc tính quan trọng
- gán trọng số cho các thuộc tính

Cám ơn !