Lecture 7

Instance-Based Learning (IBL): *k*-Nearest Neighbor e Case-Based Reasoning

Mercoledì, 17 novembre 2004

Giuseppe Manco

Readings:

Chapter 8, Mitchell

Chapter 7, Han & Kamber

Instance-Based Learning (IBL)

- **Idea**
  - Conserviamo (più o meno) tutte le istanze <x, $c(x)$>
  - Classificazione: Data una istanza x_q
  - Restituiiamo: il concetto dell'istanza più prossima nel database di prototipi
  - Giustificazione
    - *L'istanza più prossima a* x_q tende ad avere lo stesso concetto di $f(x_q)$
    - *Non vale quando i dati sono pochi*
- **Nearest Neighbor**
  - Si trovi l'esempio x_n nel training set che sia più vicino alla query x_q
  - Si effettui la stima $\hat{f}(x_q) \leftarrow f(x_n)$
- ***k*-Nearest Neighbor**
  - *f nominale*: votazione tra I *k* vicini "più vicini" a x_q
  - *f numerica:*

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} f(x_i)}{k}$$

Quando usare Nearest Neighbor

- **Proprietà ideali**
  - **Istanze in R^n**
  - **Meno di 20 attributi per instanza**
  - **Training set molto grande**
- **Vantaggi**
  - **Training veloce**
  - **Apprendimento di funzioni complesse**
  - **Non c'à perdita di informazione**
- **Svantaggi**
  - **Lento a tempo di classificazione**
  - ***Attributi irrelevanti influenzano il tutto***

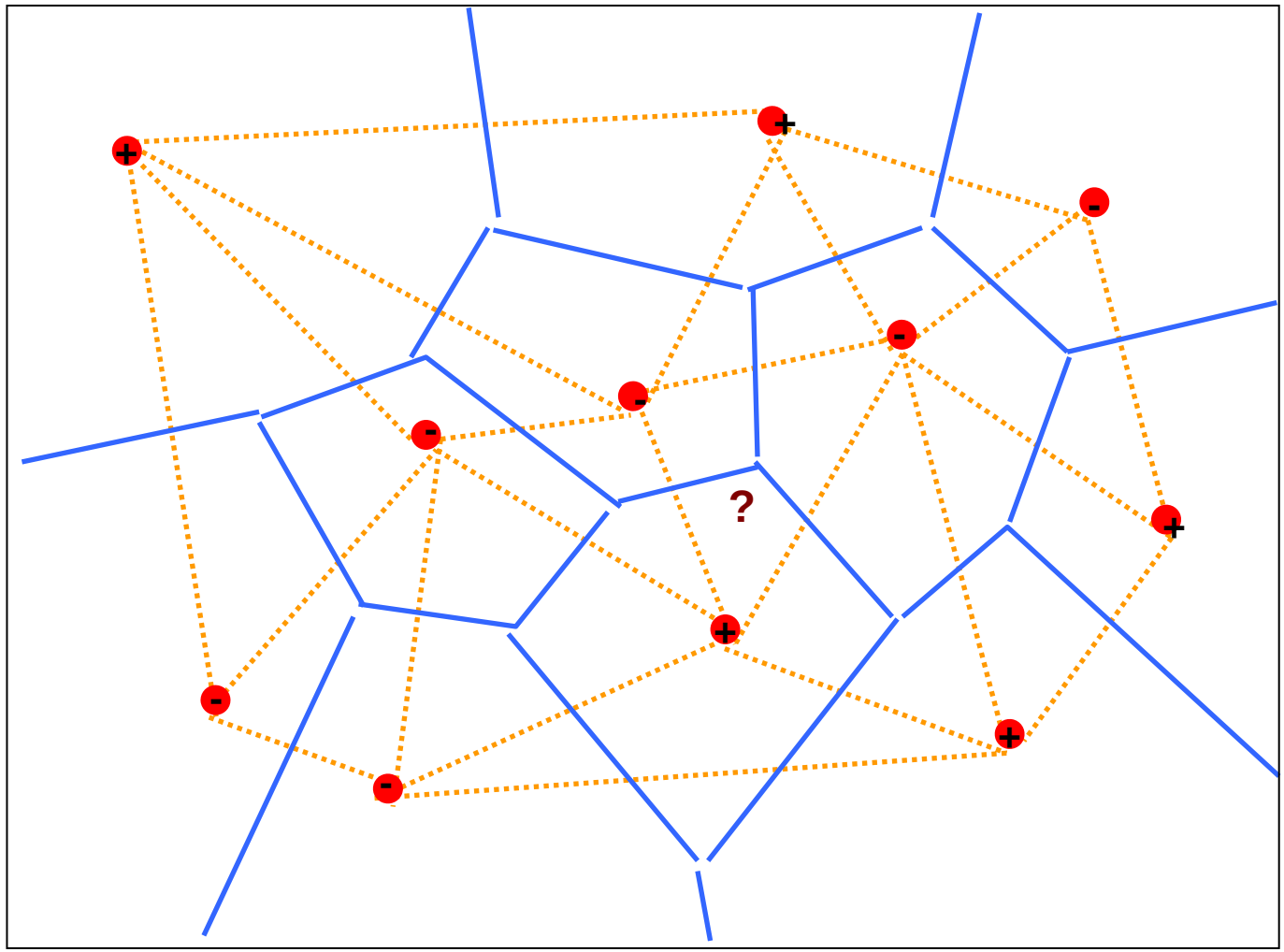
Diagrammi di Voronoi

Training Data:
Istanze etichettate

Triangolazione di Delaunay

Diagrammi di Voronoi (Nearest Neighbor)

Query
x_q



k-NN Pesato

- **Idea**
  - **Utilizziamo tutti i dati, associamo un peso ai vicini**

- **Funzione di peso**

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} w_i \cdot f(x_i)}{\sum_{i=1}^{k} w_i}$$

  - **I pesi sono proporzionali alla distanza:**

$$w_i \equiv \frac{1}{d(x_q, x_i)^2}$$

  - **$d(x_q, x_i)$ è la distanza euclidea**

Case-Based Reasoning (CBR)

- **IBL applicato a dati non numerici**
  - **Necessità di una nozione differente di “distanza”**
  - **Idea: *utilizziamo misure di similarità simbolica (sintattica)***

Esempio

Istanza	X_1	X_2	X_3
I_1	0	0	0
I_2	1	0	0
I_3	2	0	0
I_4	2.5	2	0
I_5	3	0	0
I_6	1	2	1
I_7	1.5	0	1
I_8	2	2	1
I_9	3	2	1
I_{10}	4	2	1

$$\begin{bmatrix} d(I_1,I_1) & d(I_1,I_2) & \dots & d(I_1,I_{10}) \\ d(I_2,I_1) & d(I_2,I_2) & & \cdot \\ \cdot & & & \cdot \\ \cdot & & \cdot & \cdot \\ \cdot & & & \\ d(I_{10},I_1) & d(I_{10},I_2) & & d(I_{10},I_{10}) \end{bmatrix}$$

Similarità e dissimilarità tra oggetti

- La distanza è utilizzata per misurare la similarità (o dissimilarità) tra due istanze
- Some popular ones include: Minkowski distance:

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

  - Dove $\underline{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ e $\underline{x}_j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ sono due oggetti p-dimensionali, e q è un numero primo
- se q = 1, d è la distanza Manhattan

$$d(\mathbf{x}_i,\mathbf{x}_j)=|x_{i1}-x_{j1}|+|x_{i2}-x_{j2}|+...+|x_{ip}-x_{jp}|$$

Similarità e dissimilarità tra oggetti [2]

- **se q = 2, d è la distanza euclidea:**

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+\ldots+|x_{i_p}-x_{j_p}|^2)}$$

  - **Proprietà**

$$d(\mathbf{x}_i,\mathbf{x}_j)\geq 0$$
$$d(\mathbf{x}_i,\mathbf{x}_i)=0$$
$$d(\mathbf{x}_i,\mathbf{x}_j)=d(\mathbf{x}_j,\mathbf{x}_i)$$
$$d(\mathbf{x}_i,\mathbf{x}_j)\leq d(\mathbf{x}_i,\mathbf{x}_k)+d(\mathbf{x}_k,\mathbf{x}_j)$$

- **Varianti**
  - **Distanza pesata**

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sqrt{w_1(|x_{i_1}-x_{j_1}|^2+w_2|x_{i_2}-x_{j_2}|^2+\ldots+w_p|x_{i_p}-x_{j_p}|^2)}$$

  - **Distanza Mahalanobis**

$$d(\mathbf{x}_i,\mathbf{x}_j)=\mathbf{x}_i^T\boldsymbol{\Sigma}^{-1}\mathbf{x}_j$$

Esempio

Euclidea

0	1	2	3,2	3	2,4	1,8	3	3,7	4,5
1	0	2	3,2	3	2,4	1,8	3	3,7	4,5
2	2	0	3,2	3	2,4	1,8	3	3,7	4,5
3,2	3,2	3,2	0	3	2,4	1,8	3	3,7	4,5
3	3	3	3	0	2,4	1,8	3	3,7	4,5
2,4	2,4	2,4	2,4	2,4	0	1,8	3	3,7	4,5
1,8	1,8	1,8	1,8	1,8	1,8	0	3	3,7	4,5
3	3	3	3	3	3	3	0	3,7	4,5
3,7	3,7	3,7	3,7	3,7	3,7	3,7	3,7	0	4,5
4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	0

mahalanobis

0	0,9	3,6	7,65	8,1	4,5	7,65	5,4	8,1	12,6
0,9	0	0,9	5,85	3,6	5,4	5,85	4,5	5,4	8,1
3,6	0,9	0	5,85	0,9	8,1	5,85	5,4	4,5	5,4
7,65	5,85	5,85	0	7,65	7,65	18	5,85	5,85	7,65
8,1	3,6	0,9	7,65	0	12,6	7,65	8,1	5,4	4,5
4,5	5,4	8,1	7,65	12,6	0	7,65	0,9	3,6	8,1
7,65	5,85	5,85	18	7,65	7,65	0	5,85	5,85	7,65
5,4	4,5	5,4	5,85	8,1	0,9	5,85	0	0,9	3,6
8,1	5,4	4,5	5,85	5,4	3,6	5,85	0,9	0	0,9
12,6	8,1	5,4	7,65	4,5	8,1	7,65	3,6	0,9	0

Istanza	X_1	X_2	X_3
I_1	0	0	0
I_2	1	0	0
I_3	2	0	0
I_4	2.5	2	0
I_5	3	0	0
I_6	1	2	1
I_7	1.5	0	1
I_8	2	2	1
I_9	3	2	1
I_{10}	4	2	1

manhattan

0	1	2	4,5	3	4	2,5	5	6	7
1	0	2	4,5	3	4	2,5	5	6	7
2	2	0	4,5	3	4	2,5	5	6	7
4,5	4,5	4,5	0	3	4	2,5	5	6	7
3	3	3	3	0	4	2,5	5	6	7
4	4	4	4	4	0	2,5	5	6	7
2,5	2,5	2,5	2,5	2,5	2,5	0	5	6	7
5	5	5	5	5	5	5	0	6	7
6	6	6	6	6	6	6	6	0	7
7	7	7	7	7	7	7	7	7	0

- **Similarità del coseno**
- **Similarità di Jaccard**
  - **Su dati reali**
- **Studio dell'effetto delle varie misure di similarità nei clusters**

Attributi binari

- **Distanza di Hamming**

$$d(\mathbf{x}_i, \mathbf{x}_j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|$$

  - **Distanza Manhattan quando i valori possibili sono 0 o 1**
- **In pratica, conta il numero di mismatches**

Attributi binari

- **Utilizzando la tabella di contingenza**

		Oggetto j		
		1	0	*totale*
Oggetto i	1	a	b	$a+b$
	0	c	d	$c+d$
	totale	$a+c$	$b+d$	p

- **Coefficiente di matching (invariante, se le variabili sono *simmetriche*):**

$$d(i, j) = \frac{b + c}{a + b + c + d}$$

- **Coefficiente di Jaccard (noninvariante se le variabili sono *asimmetriche*):**

$$d(i, j) = \frac{b + c}{a + b + c}$$

Dissimilarità tra attributi binari

- **Esempio**

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

  - gender è simmetrico
  - Tutti gli altri sono asimmetrici
  - Poniamo Y e P uguale a 1, e N a 0

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

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

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

Indici di similarità demografica

- Proporzionale al numero di “uni” comuni (1-1)
- Inversamente proporzionale agli 0-1 e ai 1-0
- Non affetta da 0-0
- Indice di Condorcet =
  - $a / (a + {}^{1}/_{2}(b + c))$
- Indice di Dice =
  - $a / (a + {}^{1}/_{4}(b + c))$
- Dice meno efficace di Condorcet
  - Appropriato su oggetti estremamente differenti

Variabili Nominali

- **Generalizzazione del meccanismo di variabili binarie**
- **Metodo 1: Matching semplice**
  - ***m*: # di matches, *p*: # di attributi nominali**

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

- **metodo 2: binarizzazione**

Esercizio: PlayTennis

Day	Outlook	Temperature	Humidity	Wind	*PlayTennis*?
1	Sunny	Hot	High	Light	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Light	Yes
4	Rain	Mild	High	Light	Yes
5	Rain	Cool	Normal	Light	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Light	No
9	Sunny	Cool	Normal	Light	Yes
10	Rain	Mild	Normal	Light	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Light	Yes
14	Rain	Mild	High	Strong	No

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	Strong	?

Playtennis [2]

- d(I,d1) = 2/4 = 0.5
- d(I,d2) = 1/4 = 0.25
- d(I,d3) = 3/4 = 0.75
- d(I,d4) = 3/4 = 0.75
- d(I,d5) = 3/4 = 0.75
- d(I,d6) = 2/4 = 0.5
- d(I,d7) = 2/4 = 0.75
- d(I,d8) = 2/4 = 0.5
- d(I,d9) = 2/4 = 0.5
- d(I,d10) = 3/4 = 0.75
- d(I,d11) = 2/4 = 0.5
- d(I,d12) = 2/4 = 0.5
- d(I,d13) = 4/4 = 1
- d(I,d14) = 2/4 = 0.5

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	Strong	?

Day	Outlook	Temperature	Humidity	Wind	*PlayTennis*?
1	Sunny	Hot	High	Light	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Light	Yes
4	Rain	Mild	High	Light	Yes
5	Rain	Cool	Normal	Light	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Light	No
9	Sunny	Cool	Normal	Light	Yes
10	Rain	Mild	Normal	Light	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Light	Yes
14	Rain	Mild	High	Strong	No

Lazy e Eager Learning

- **Lazy Learning**
  - **Nessuna generalizzazione: aspettiamo la query**
    - ***k*-nearest neighbor (*k*-NN)**
    - **Case-based reasoning (CBR)**
- **Eager Learning**
  - **generalizzazione**
    - **ID3, backpropagation, simple (Naïve) Bayes, etc.**
- **Qual'è la differenza?**
  - **Eager crea un'approssimazione globale**
  - **Lazy può creare molte approssimazioni locali**