K Nearest Neighbors

Sudipta Roy

KNN - Definition

KNN is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure

KNN – different names

- K-Nearest Neighbors
- Memory-Based Reasoning
- Example-Based Reasoning
- Instance-Based Learning
- Case-Based Reasoning
- Lazy Learning

KNN – Short History

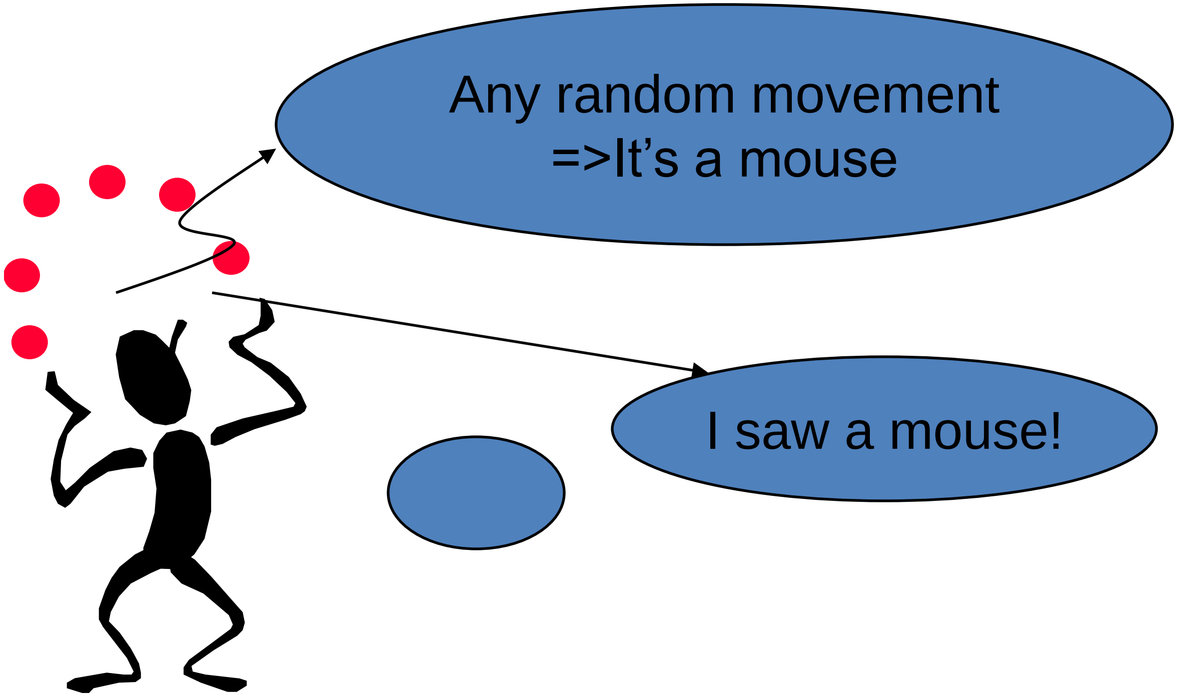
- Nearest Neighbors have been used in statistical estimation and pattern recognition already in the beginning of 1970's (non-parametric techniques).
- Dynamic Memory: A theory of Reminding and Learning in Computer and People (Schank, 1982).
- People reason by remembering and learn by doing.
- Thinking is reminding, making analogies.
- Examples = Concepts???

Different Learning Methods

- Eager Learning
  - Explicit description of target function on the whole training set
- Instance-based Learning
  - Learning=storing all training instances
  - Classification=assigning target function to a new instance
  - Referred to as “Lazy” learning

Different Learning Methods

- Eager Learning



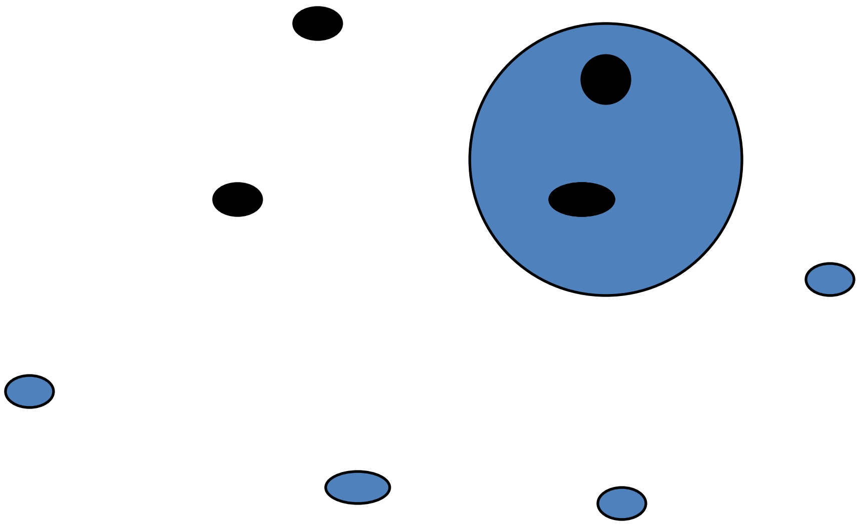
Any random movement
=>It's a mouse

I saw a mouse!

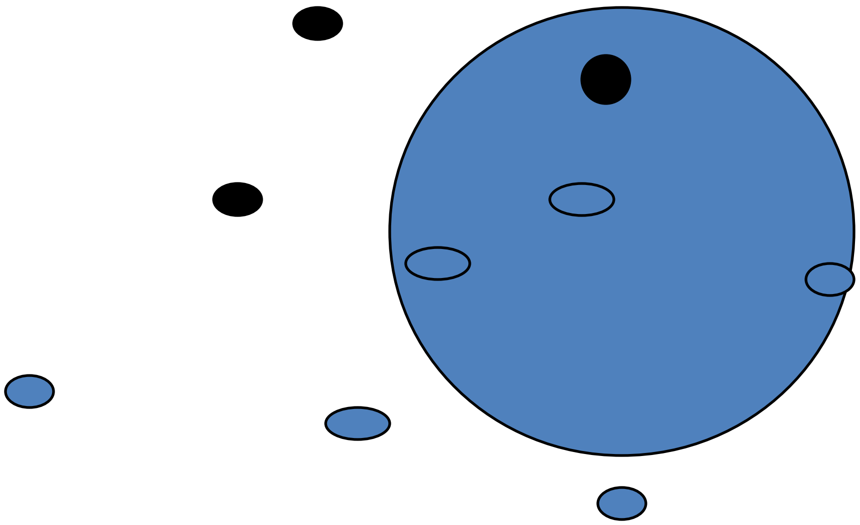
Instance-based Learning

Its very similar to a Desktop!!

1-Nearest Neighbor



3-Nearest Neighbor



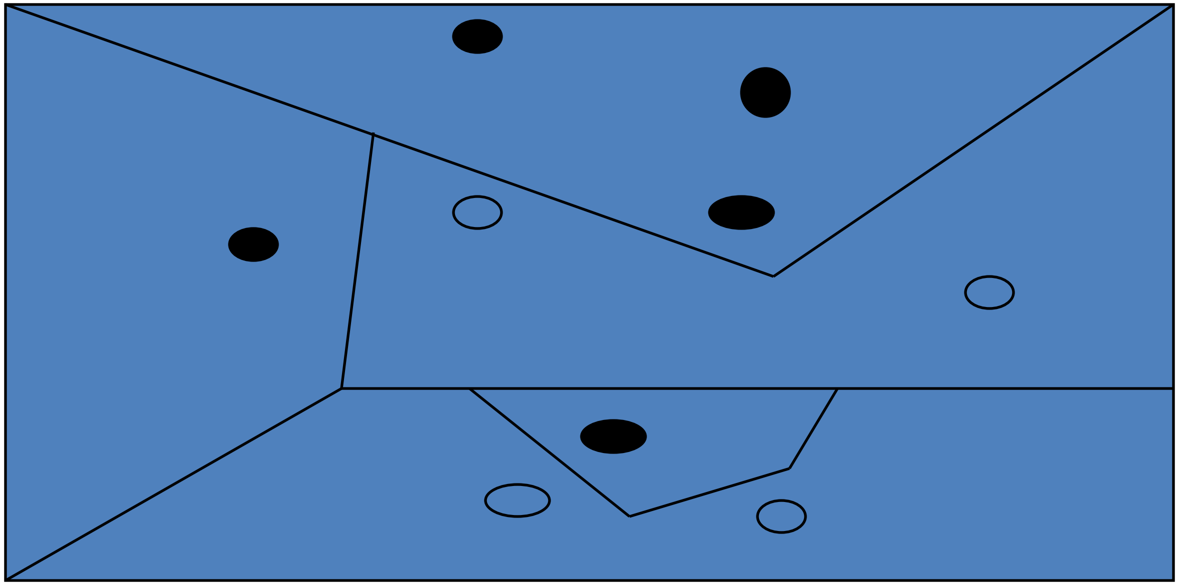
K-Nearest Neighbor

- An arbitrary instance is represented by (a$_1$(x), a$_2$(x), a$_3$(x),.., a$_n$(x))
  - a$_i$(x) denotes features
- Euclidean distance between two instances

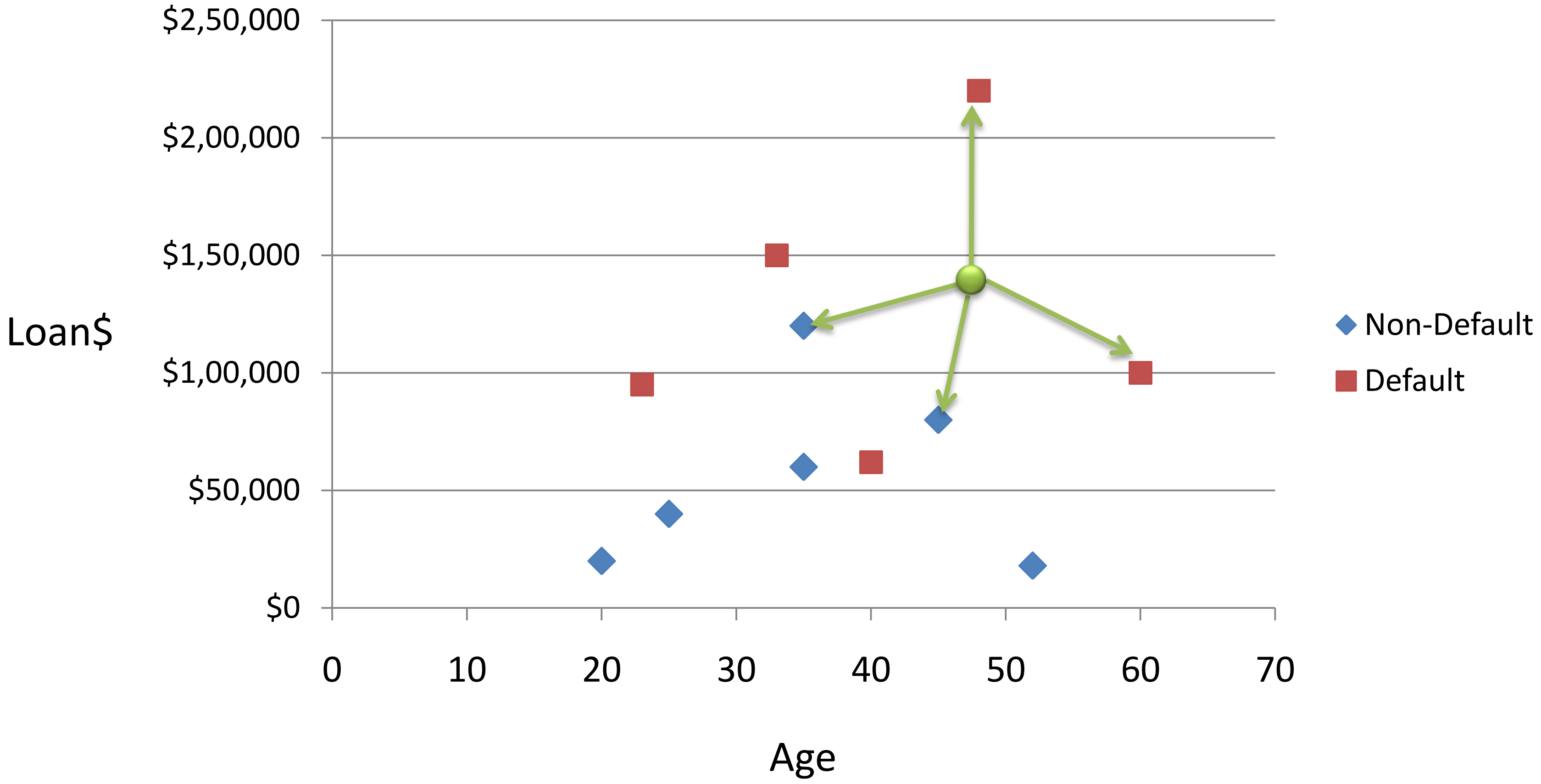
  d(x$_i$, x$_j$)=sqrt (sum for r=1 to n (a$_r$(x$_i$) - a$_r$(x$_j$))^2)
- Continuous valued target function
  - mean value of the k nearest training examples

Voronoi Diagram

- Decision surface formed by the training examples



KNN Classification



Age	Loan	Default
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
40	$62,000	Y
60	$100,000	Y
48	$220,000	Y
33	$150,000	Y
48	**$142,000**	**?**

KNN Classification – Distance

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
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

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

KNN Classification – Standardized Distance

Age	Loan	Default
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

Age	Loan	Default	Distance
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

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

KNN Regression - Distance

Age	Loan	House Price Index	Distance
25	$40,000	135	102000
35	$60,000	256	82000
45	$80,000	231	62000
20	$20,000	267	122000
35	$120,000	139	22000
52	$18,000	150	124000
23	$95,000	127	47000
40	$62,000	216	80000
60	$100,000	139	42000
48	$220,000	250	78000
33	$150,000	264	8000
48	**$142,000**	**?**	

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

KNN Regression – Standardized Distance

Age	Loan	House Price Index	Distance
0.125	0.11	135	0.7652
0.375	0.21	256	0.5200
0.625	0.31	231	0.3160
0	0.01	267	0.9245
0.375	0.50	139	0.3428
0.8	0.00	150	0.6220
0.075	0.38	127	0.6669
0.5	0.22	216	0.4437
1	0.41	139	0.3650
0.7	1.00	250	0.3861
0.325	0.65	264	0.3771
0.7	**0.61**	**?**	

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

KNN – Number of Neighbors

- If K=1, select the nearest neighbor
- If K>1,
  - For classification select the most frequent neighbor.
  - For regression calculate the average of K neighbors.

KNN - Applications

- Classification and Interpretation
  - legal, medical, news, banking
- Problem-solving
  - planning, pronunciation
- Function learning
  - dynamic control
- Teaching and aiding
  - help desk, user training

Summary

- KNN is conceptually simple, yet able to solve complex problems
- Can work with relatively little information
- Learning is simple (no learning at all!)
- Memory and CPU cost
- Feature selection problem
- Sensitive to representation

Questions?