Artificial Intelligence 2

Classification

- Given some pre-classified data, classify new data

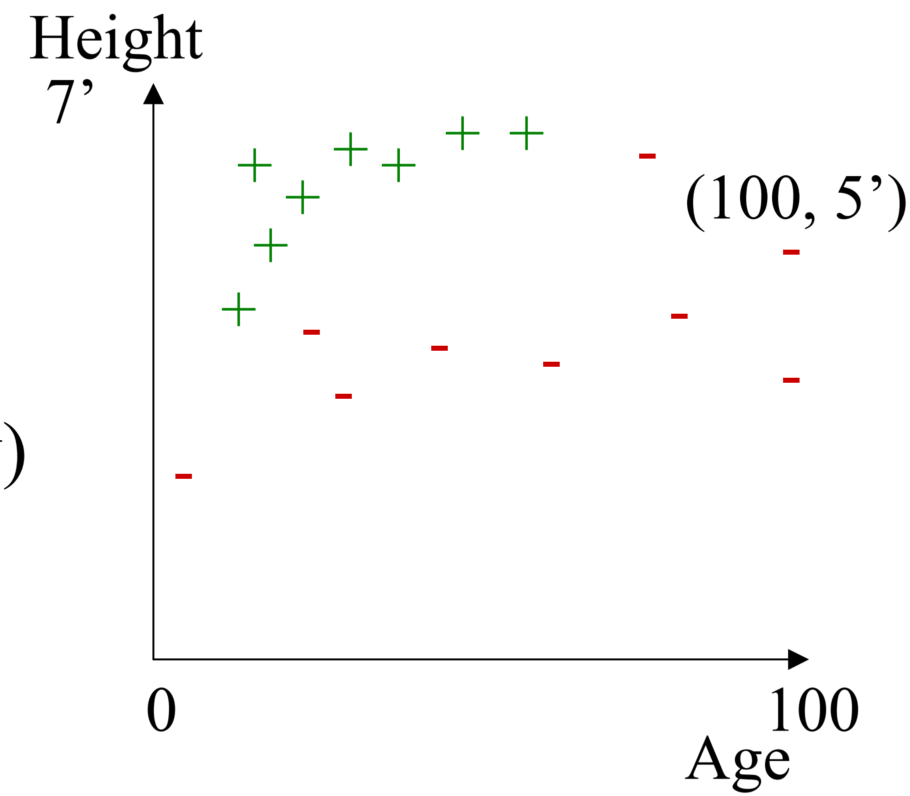
Example:

Ask 20 people their age, height and if they like to play basketball

Ask another person their age and height. Based on the information you just learned about preferences of people in his/her age and height group, attempt to sell this person tickets to a basketball game

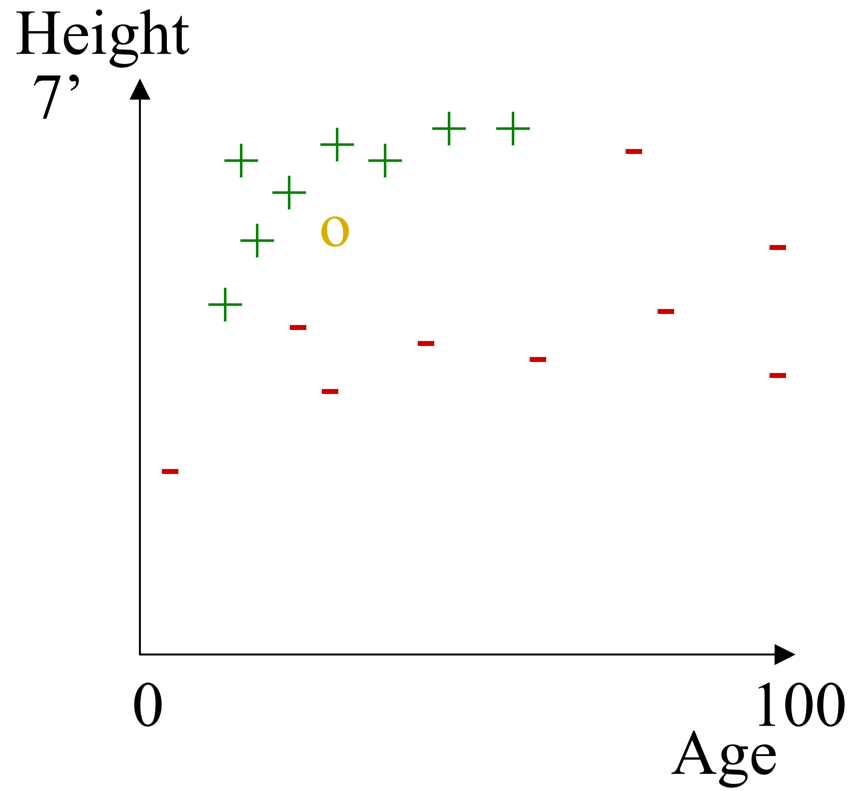
Classification

- Let the x-dimension be "age"
- Let the y-dimension be "height"
- Whenever a person with age x and height y wants to play basketball, plot a "+" at (x, y)
- Whenever a person with age x and height y does not want to play basketball, plot a "-" at (x, y)



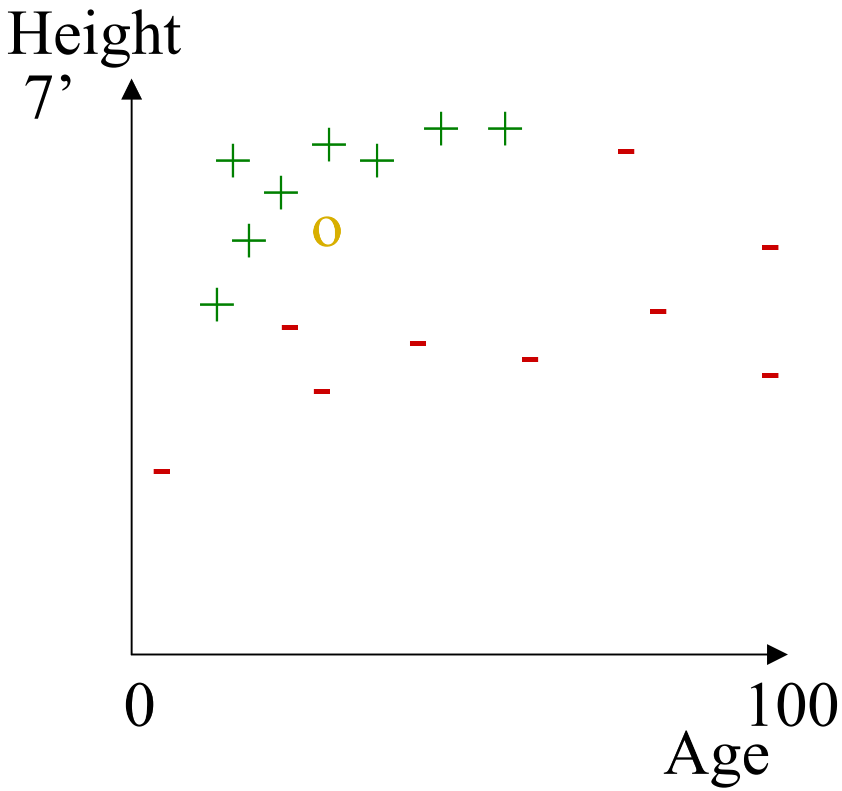
Classifying New Data

- You have surveyed a new person with age *a* and height *h*.
- Plot a "o" in the age-height space at (*a*,*h*)
- Should you sell basketball tickets to this person?



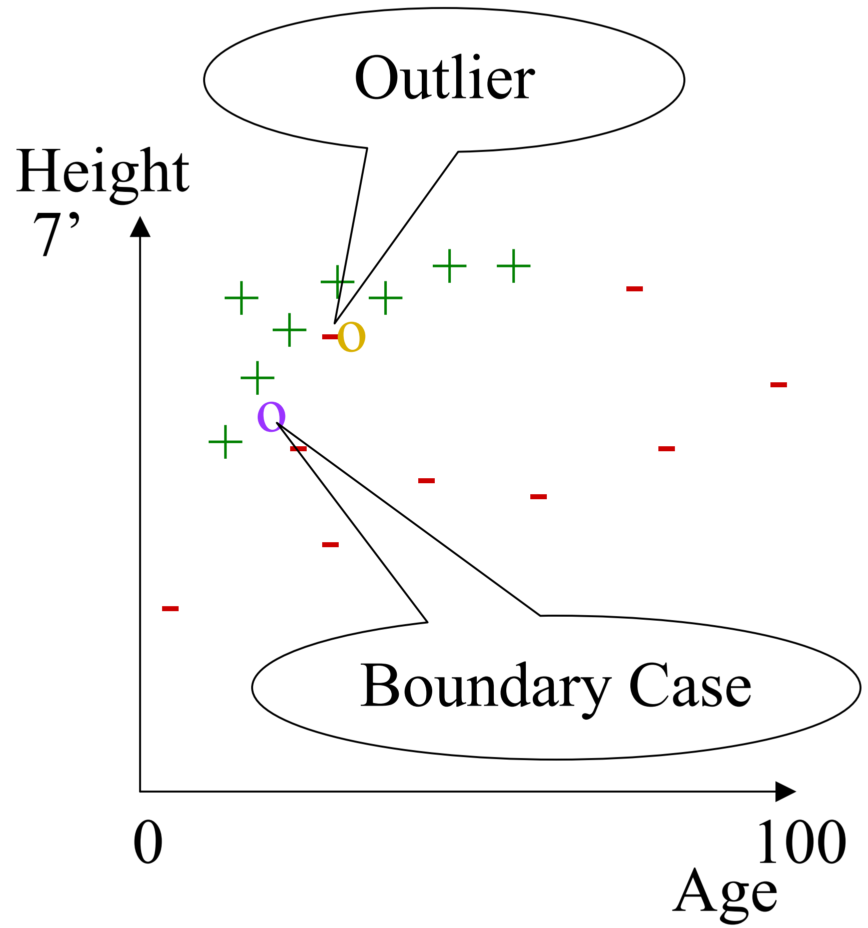
Nearest Neighbour Classification

- Classify "o" same as its "nearest neighbour"
- The nearest neighbour of "o" is one that has the smallest *distance* to "o".
- *Euclidean Distance* between $p = (x_p, y_p)$ and $q = (x_q, y_q)$ is $\sqrt{(x_p - x_q)^2 + (y_p - y_q)^2}$



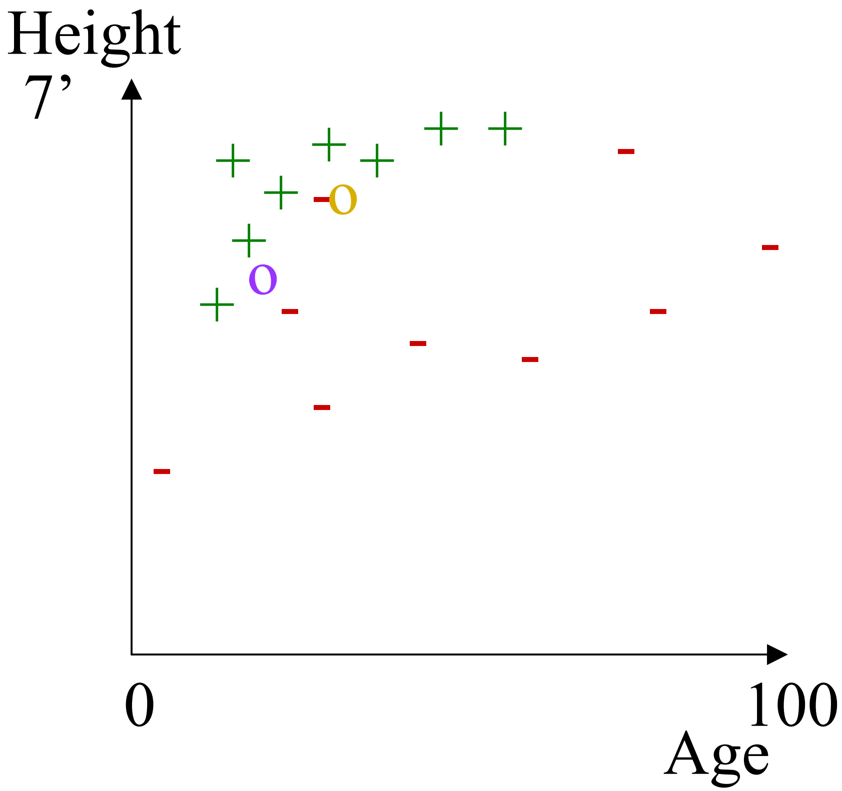
Nearest Neighbour Classification

- Outlier: a statistical anomaly
- How should "o" be classified?
- What about "o"?



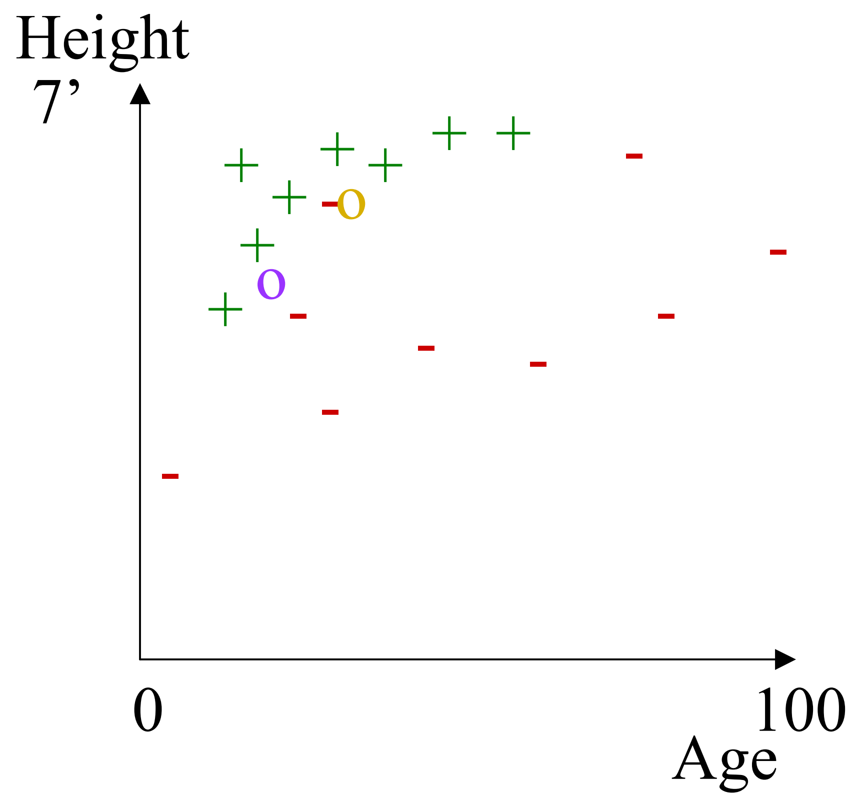
k-Nearest Neighbours

- Consider the *k* nearest neighbours of a point *p* to be classified.
- Assign *p* the class to which most of the *k* nearest neighbours belong
- Possible values for *k*: 3, 5, 7, … Why?



k-Nearest Neighbours

- Choosing *k*:
  - What kind of problem occurs when *k* is too small?
  - What kind of problem occurs when *k* is too big?
- Choosing a good *k*:
  - Cross-Validation
    - leave out some classified data
    - find a *k* that gives the best result



k-Nearest Neighbours

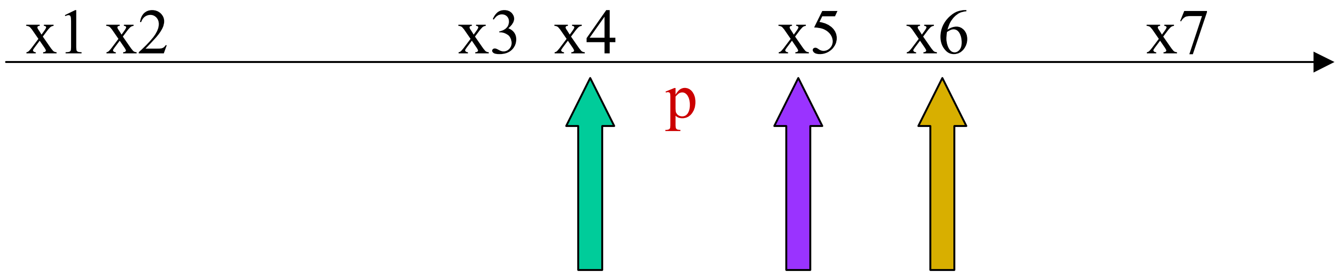
- What are possible conceptual problems?

  Is age as significant as height (vice versa)?

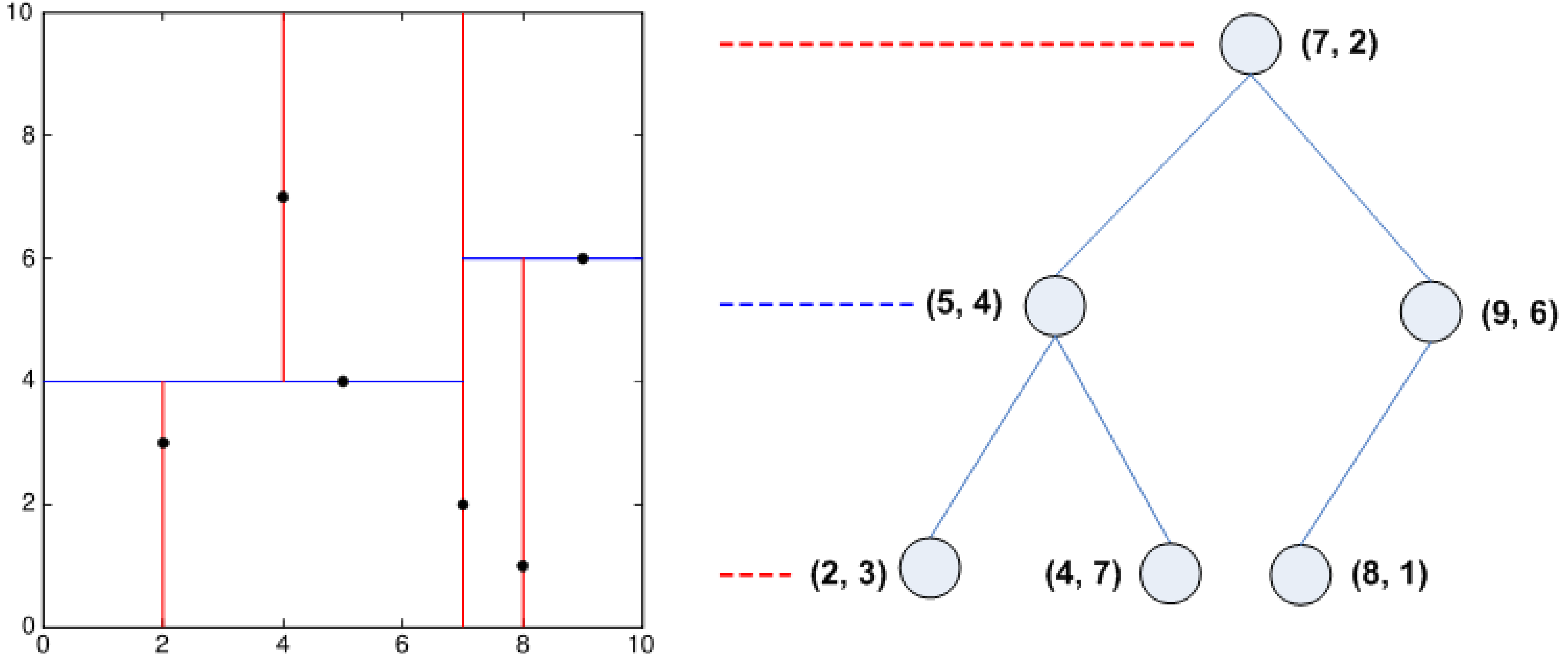
- How much computation is done?
  - For each of *n* pre-classified points, compute the distance to them from *p* and record closest point found
  - Look at ALL the points!

Space Partitioning

- Sequential Search vs. Binary Search
- Suppose only had 1 dimension:

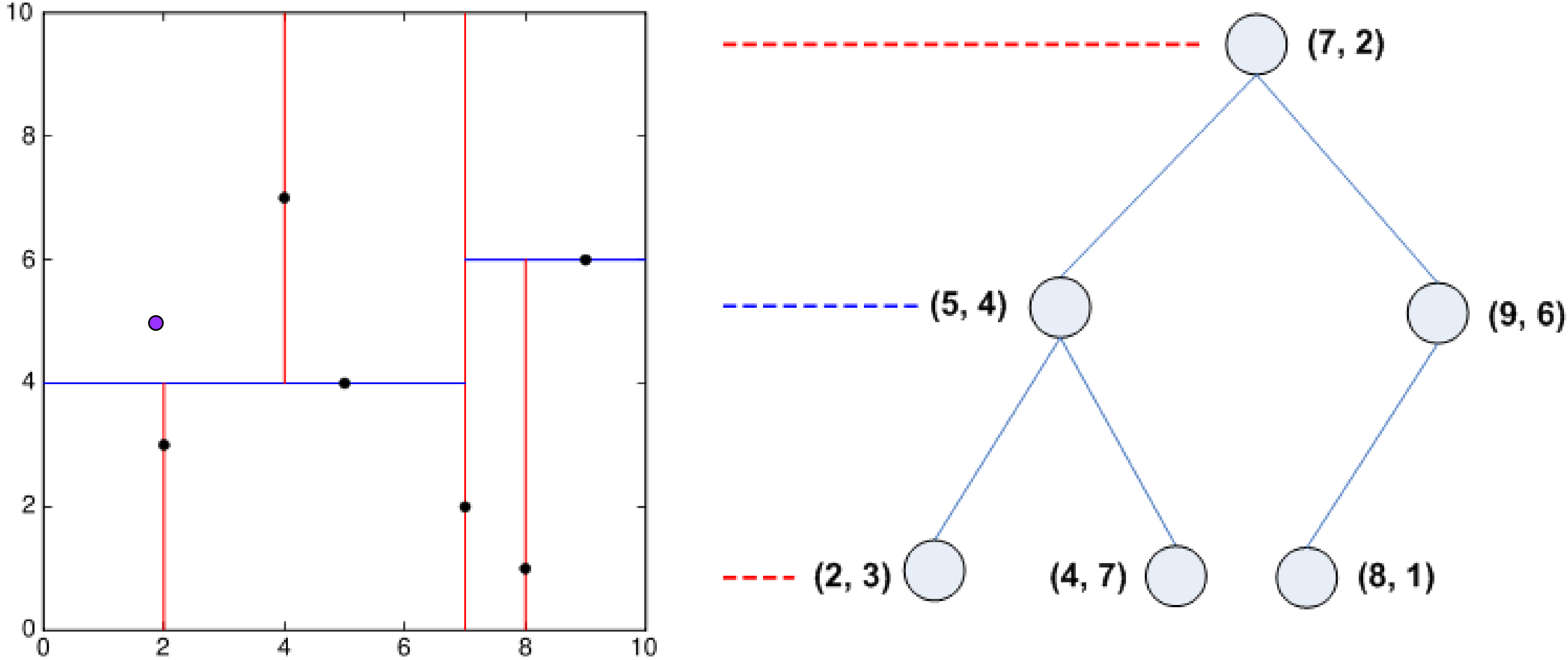


2 Dimensions



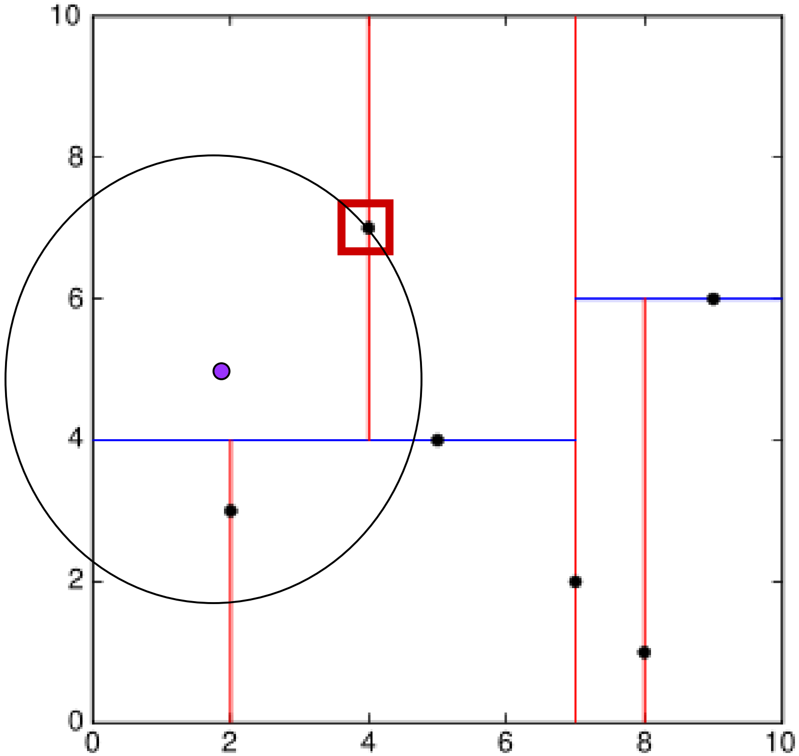
Alternate *x* and *y* slices to divide space along the *midpoint*

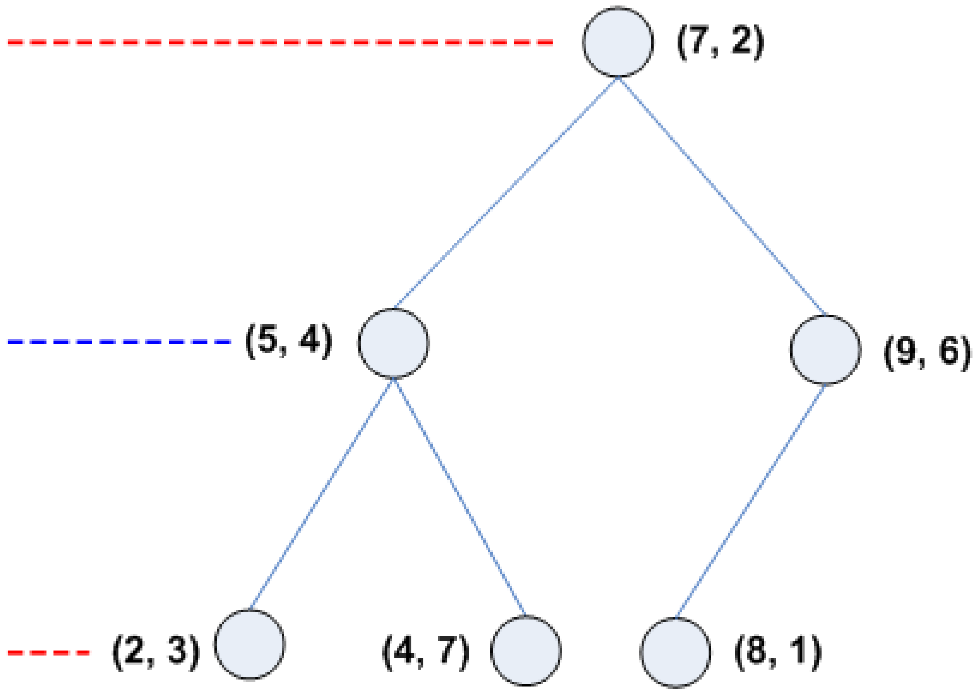
Faster Nearest Neighbour Search



- Find nearest neighbour to (2, 5)
- First guess is (4, 7)

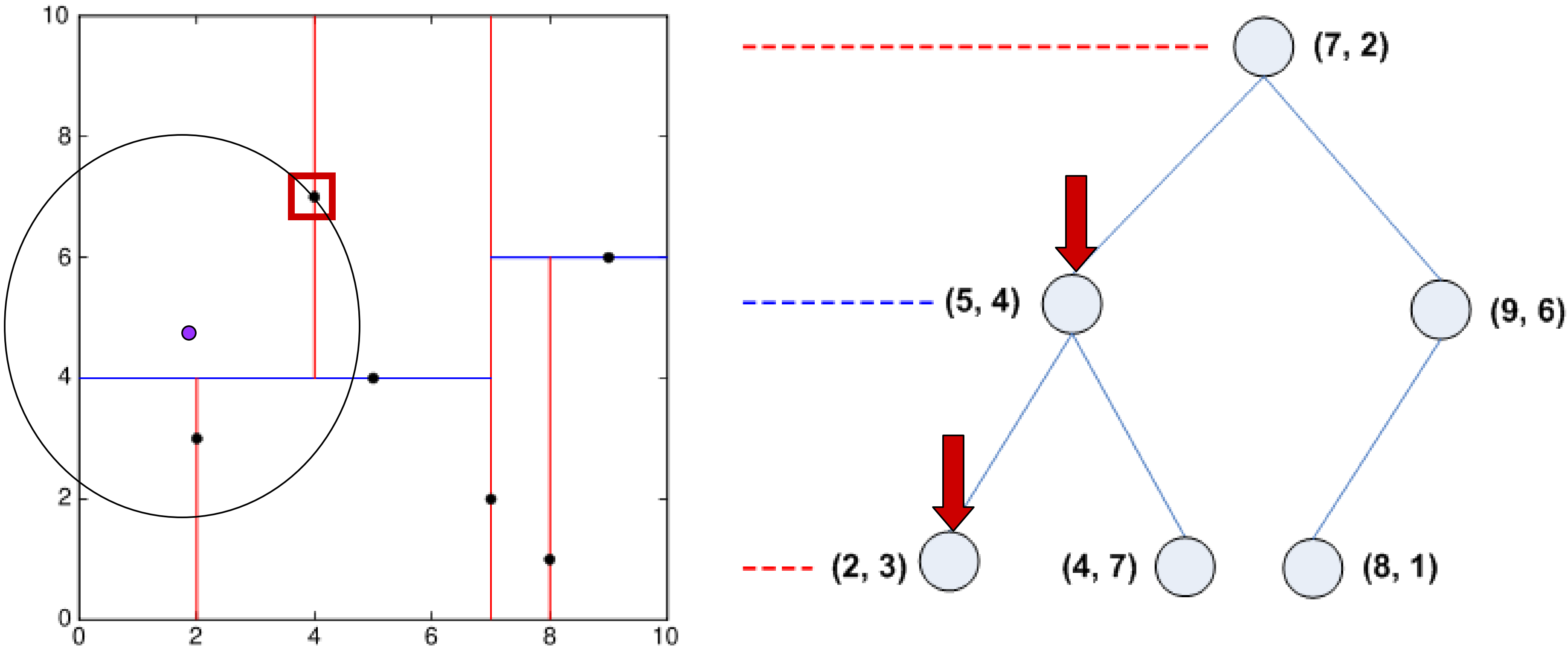
Faster Nearest Neighbour Search





- Candidate nearest neighbour: □
- Need to confirm that □ is indeed the closest, i.e. that no other point lies within the circle around the purple point

Faster Nearest Neighbour Search



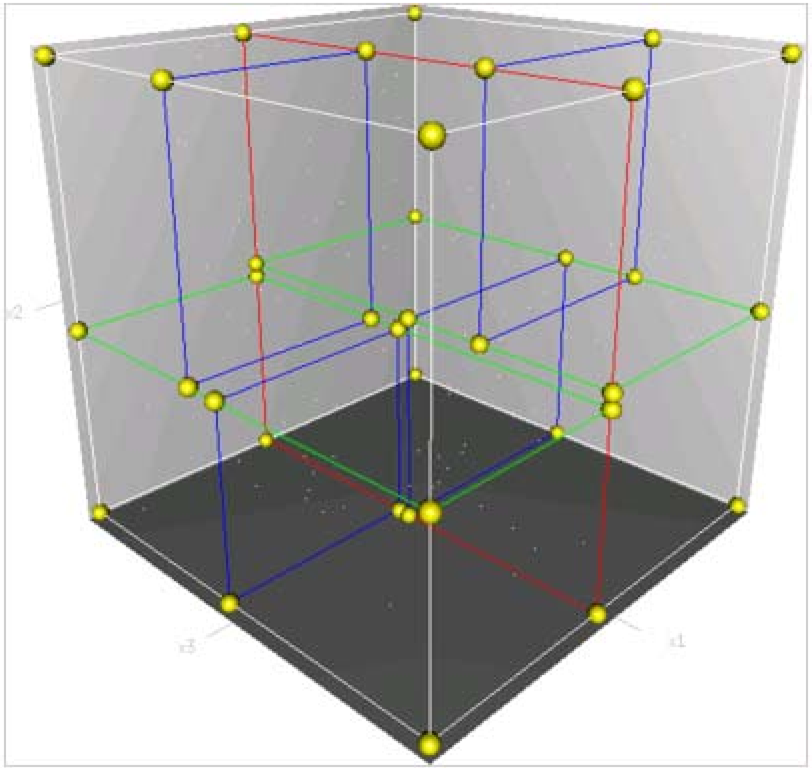
- Go back up in the *kd*-tree to test which points could lie within the circle (closer to the purple point than the red square)

kd-trees

- Moral of the story:
  - Could possibly save time by using a *kd*-tree *data structure*
  - Building this data structure is an overhead
  - In general, this problem is very difficult! (although has a relatively simple formulation)

kd-tree

- Suppose you asked *k* questions before you asked about basketball
- Then each person would be a point in *k*-dimensional space!!!



Back to AI…

- Recall that we are building rational agents that can perform certain (specialized) actions *in a clever way* (supplementing or augmenting human intelligence)

- Now that able to classify novel data from examples, consider the following *intelligent* scenario

Game Theory

	Prisoner B Stays Silent	Prisoner B Betrays
Prisoner A Stays Silent	Both serve six months	Prisoner A serves ten years Prisoner B goes free
Prisoner A Betrays	Prisoner A goes free Prisoner B serves ten years	Both serve two years

- What logical choice should each prisoner make to minimize his/her time in prison?

Game Theory

	Prisoner B Stays Silent	Prisoner B Betrays
Prisoner A Stays Silent	Both serve six months	Prisoner A serves ten years Prisoner B goes free
Prisoner A Betrays	Prisoner A goes free Prisoner B serves ten years	Both serve two years

A	B silent	B betrays
A silent	.5	10
A betrays	0	2

B	B silent	B betrays
A silent	.5	0
A betrays	10	2

Game Theory

Dominant strategy: both A and B betray.

If they could communicate, they would both be silent.

Prisoner's Dilemma one of most cited examples of Game Theory

A	B silent	B betrays
A silent	.5	10
A betrays	0	2

B	B silent	B betrays
A silent	.5	0
A betrays	10	2

"Battle of the Sexes"

	Opera	Football
Opera	3,2	0,0
Football	0,0	2,3

- Numbers represent happiness ☺
- When do different actions, unhappy
- When do same action, one more happy than other
- Again make choice independently
- No dominant strategy!

Matching Pennies

	Heads	Tails
Heads	+1, -1	-1, +1
Tails	-1, +1	+1, -1

- Either win or lose $1
- Each player *chooses* head or tail
- If player A and player B choose same thing, player A gets $1; otherwise, player B gets $1
- What's the optimal strategy?

Matching Pennies

	Heads	Tails
Heads	+1, -1	-1, +1
Tails	-1, +1	+1, -1

- No clear optimal strategy like before
- Best strategy: choose *heads* or *tails* with equal probability at random
- Why must the probability be equal?

John Nash

- John Nash showed that in any game there exists at least one strategy such that all players would be happy playing one of these strategies without wanting to switch to any other strategy (so called *Nash Equilibrium)*.