 The University of Auckland

## Case-Based Reasoning

---

Lazy Learning  
Prof. Ian Watson

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---


---

---

---

---

---

 The University of Auckland 2

## Eager Learning

- ML algorithms like ID3, C4.5 or Neural Networks are *eager* learners
  - Use a training data set to
  - Generalize rules, induce a tree or a function that can be applied to categorize future inputs
  - Processing time is done up-front before query time
  - After querying they discard any inputs

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---


---

---

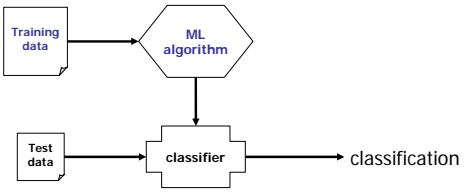
---

---

---

 The University of Auckland 3

## Eager Learning



```

    graph TD
      TD[Training data] --> MA{{ML algorithm}}
      MA --> C[classifier]
      TeD[Test data] --> C
      C --> Cl[classification]
  
```

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

4



## Eager learning methodology

- Obtain data set
- Identify target (output) attribute (this is what we want to predict)
- Analyse input features
  - Estimate which are predictive of target
  - Are combinations of input features required (eg a simple ratio of two inputs)
- Analyse data set and remove noisy items
- Divide data set in training and test sets

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

5



## Eager learning methodology

- Identify possible ML algorithms based on:
  - Data types (discrete, continuous)
  - Classification or regression task
  - Type of output required
    - Function
    - Decision tree
    - Neural network

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

6



## Eager learning methodology

- Run algorithm(s) on training data
- Validate on test data
- Better still do 10 fold cross validation
- Tweak parameters of algorithm
- Repeat validation
- Consider using an ensemble of algorithms

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

7



## Lazy learners

- Lazy learners have three characteristics:
  - They defer all (most) processing until query/run-time
  - They discard any generated functions/answers
  - They retain the query with the stored data

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

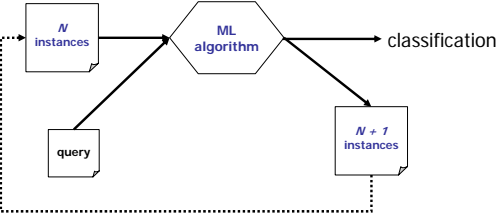
---

---

8



## Lazy Learning



The diagram illustrates the Lazy Learning process. It starts with a box labeled 'N instances' and a box labeled 'query'. Both have arrows pointing to a central hexagon labeled 'ML algorithm'. From the 'ML algorithm', an arrow points to the word 'classification'. Another arrow points from the 'ML algorithm' to a box labeled 'N + 1 instances'. A dashed line connects the 'N + 1 instances' box back to the 'query' box, indicating that the query is stored for future use.

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

9



## Lazy vs. Eager

- Lazy learners have low computational costs at training ( $\sim 0$ )
- But may have high storage costs
- High computational costs at query
- Lazy learners can respond well to dynamic data where it would be necessary to constantly re-train an eager learner

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

10



## Instance-based learners

- store *all* the training data
- when a new query instance is encountered, a set of related instances are retrieved from memory and used to classify the instance
- can construct a different approximation function of the target function for each distinct query instance

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

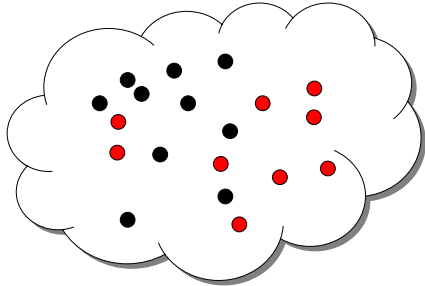
---

---

11



## Eager learners



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

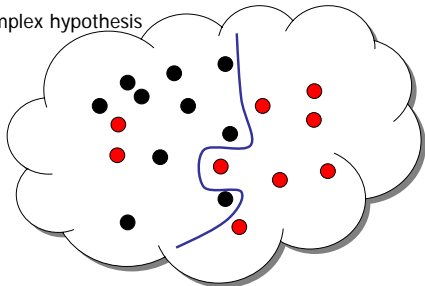
---

12



## Eager learners

A complex hypothesis



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

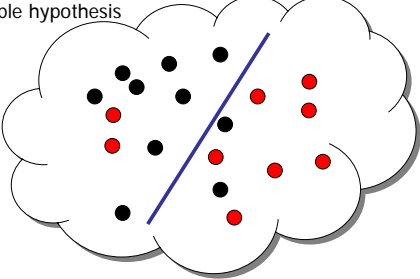
---

13



## Eager learners

A simple hypothesis



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

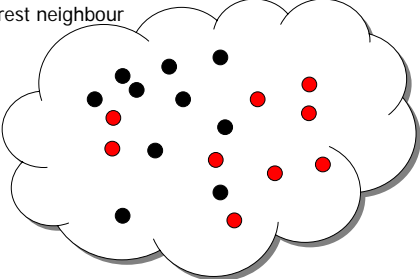
---

14



## Lazy learners

k-nearest neighbour



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

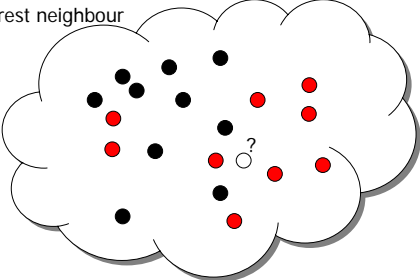
---

15



## Lazy learners

k-nearest neighbour



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

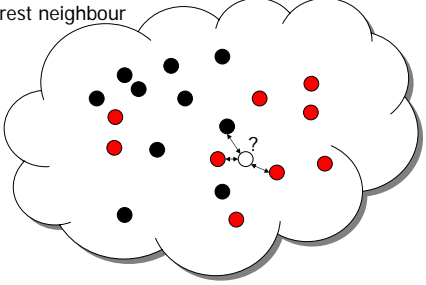
---

16



## Lazy learners

k-nearest neighbour



© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

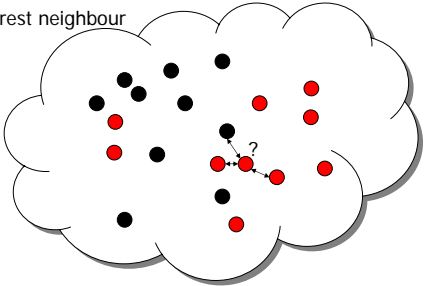
---

17



## Lazy learners

k-nearest neighbour



© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

18



## Instance-based learners

- significant advantage
- when the target function is potentially very complex
- but can be described by a collection of simple local approximations

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

19



## Instance-based learners

- Disadvantages
  - cost of classifying new instances can be high, so efficiently indexing training instances very important
  - Similarity has to be determined for each attribute or feature

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---

---

---


---

---

---

---

20



## Instance-based learners

- Nearest neighbour (k-NN)
  - most basic method - all instances are points in an  $n$ -dimensional space
  - distance is defined as standard Euclidean distance
  - K-NN finds the *nearest* neighbours to a query in the  $n$ -dimensional space
  - values may be discrete or real

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---

---

---


---

---

---

---

21



## Nearest Neighbour

$$\text{Similarity}(T, S) = \sum_{i=1}^n f(T_i, S_i) \times w_i$$

where:  
 $T$  is the target case  
 $S$  is the source case  
 $n$  is the number of attributes in each case  
 $i$  is an individual attribute from 1 to  $n$   
 $f$  is a similarity function for attribute  $i$  in cases  $T$  and  $S$  and  
 $w$  is the importance weighting of attribute  $i$

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

---


---

22

The University of Auckland

## Nearest Neighbour

- imagine a decision with two factors that influence it
- should you grant a person a loan?
  - net monthly income
  - monthly loan repayment



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

---

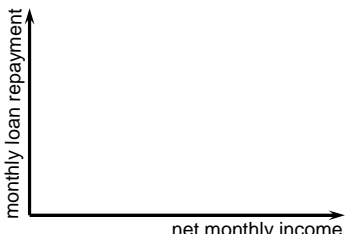
---

23

The University of Auckland

## Nearest Neighbour

- these factors can be used as axes for a graph



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

---

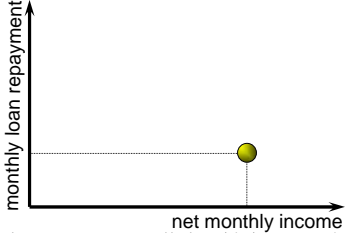
---

24

The University of Auckland

## Nearest Neighbour

- a previous loan can be plotted against these axes



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

---

---



25

The University of Auckland

## Nearest Neighbour

- and a second loan

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

26

The University of Auckland

## Nearest Neighbour

- and more loans

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

27

The University of Auckland

## Nearest Neighbour

- and even more loans

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

28

The University of Auckland

## Nearest Neighbour

- past cases (loans) may form clusters

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

29

The University of Auckland

## Nearest Neighbour

- past cases (loans) may tend to form clusters

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

30

The University of Auckland

## Nearest Neighbour

- past cases (loans) may tend to form clusters

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

31

The University of Auckland

## Nearest Neighbour

- a new loan prospect can be plotted on the graph

A scatter plot with 'monthly loan repayment' on the vertical axis and 'net monthly income' on the horizontal axis. There are two distinct clusters of data points. The upper-left cluster consists of five red circular points. The lower-right cluster consists of five yellow-green circular points. The plot is empty, indicating that no new case has been added yet.

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

32

The University of Auckland

## Nearest Neighbour

- a new loan prospect can be plotted on the graph

The same scatter plot as in slide 31, but with a new data point added. This point is cyan and is located within the lower-right cluster of yellow-green points. An arrow points from the text 'new case' to this cyan point.

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

33

The University of Auckland

## Nearest Neighbour

- and the distance to its nearest neighbours calculated

The same scatter plot as in slide 32, showing the cyan 'new case' point. An arrow originates from the cyan point and points towards the nearest red point in the upper-left cluster, representing the calculation of distance to a neighbor.

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

34

The University of Auckland

## Nearest Neighbour

- and the distance to its nearest neighbours calculated

monthly loan repayment

net monthly income

© University of Auckland www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

35

The University of Auckland

## Nearest Neighbour

- and the distance to its nearest neighbours calculated

monthly loan repayment

net monthly income

© University of Auckland www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

36

The University of Auckland

## Nearest Neighbour

- the best matching past case is the closest

© University of Auckland www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

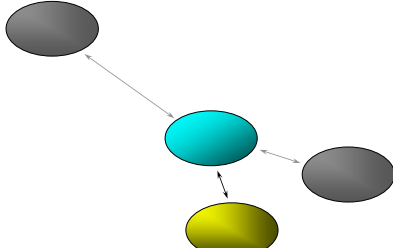
---

37

The University of Auckland

## Nearest Neighbour

- the best matching past case is the closest



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

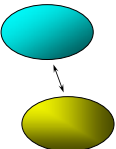
---

38

The University of Auckland

## Nearest Neighbour

- this suggests a precedent



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

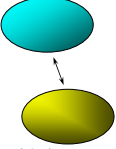
---

39

The University of Auckland

## Nearest Neighbour

- this suggests a precedent
- the loan will be successful



© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

40

The University of Auckland

## Nearest Neighbour

- over time the prediction can be validated

monthly loan repayment

net monthly income

© University of Auckland www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

41

The University of Auckland

## Nearest Neighbour

- over time the prediction can be validated

monthly loan repayment

net monthly income

it was a good loan

© University of Auckland www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

42

The University of Auckland

## Nearest Neighbour

- the system is learning to differentiate good and bad loans better

monthly loan repayment

net monthly income

© University of Auckland www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---

43

The University of Auckland

## Nearest Neighbour

- as more cases are acquired its performance improves

monthly loan repayment

net monthly income

© University of Auckland [www.cs.auckland.ac.nz/~ian/](http://www.cs.auckland.ac.nz/~ian/) [ian@cs.auckland.ac.nz](mailto:ian@cs.auckland.ac.nz)

---

---

---

---

---

---

---

---

44

The University of Auckland

## Nearest Neighbour

loan repayment

monthly income

case B

case A

case T

Euclidean Distance  
 $X_B^2 + Y_B^2 = H_B^2$

$X_B$

$Y_B$

$X_A$

© University of Auckland [www.cs.auckland.ac.nz/~ian/](http://www.cs.auckland.ac.nz/~ian/) [ian@cs.auckland.ac.nz](mailto:ian@cs.auckland.ac.nz)

---

---

---

---

---

---

---

---

45

The University of Auckland

## Nearest Neighbour

The weight of the X axis (income) is increased

loan repayment

monthly income

case B

case A

case T

$X_B$

$Y_B$

$X_A$

© University of Auckland [www.cs.auckland.ac.nz/~ian/](http://www.cs.auckland.ac.nz/~ian/) [ian@cs.auckland.ac.nz](mailto:ian@cs.auckland.ac.nz)

---

---

---

---


---

---

---

---

46



## Nearest Neighbour

- Require a unique similarity function for each attribute or feature (not always a trivial problem) – *local similarity*  $f(T_i, S_j)$
- Local similarities are combined to give a *global similarity* –  $\text{sim}(T, S)$
- k-NN Requires every feature of the query to be compared to every feature of every instance/case at run-time
- Not very efficient ☹

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---

---

---


---

---

---

---

47



## Nearest Neighbour

- distance weighted k-Nearest neighbour is a highly effective algorithm for many practical problems robust to noisy data if the training set is large enough
- bias is that the classification of an instance is most similar to other instances that are nearby in Euclidean distance
- But then again that's the point

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---

---

---


---

---

---

---

48



## Nearest Neighbour

- because distance is calculated on all attributes - irrelevant attributes are a problem - curse of dimensionality
- some approaches weight attributes to overcome this - stretching the Euclidean space – determined automatically using cross-validation
- alternatively eliminate the least relevant attributes - they used leave-one out cross-validation – ideal for IBL

© University of Auckland      www.cs.auckland.ac.nz/~ian/      ian@cs.auckland.ac.nz

---

---

---

---

---

---

---


---

---

---



49



## Nearest Neighbour

- could locally stretch an axis...but more degrees of freedom...so more chance of overfitting...useful if problem space is not uniform...problem of over fitting
- much less common, but it is used in CBR
- efficient indexing of instances can be done with kd-trees (we'll discuss later)
- possible to pre-compute a position of each instance in the Euclidean space then simply position query in the space

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---


---

---

---

---

50



## Summary

- IBLs (k-NN is an IBL) delay processing until prediction time they form a different local approximation for each query instance
- can model complex functions by a combination of less-complex local approximations
- information present in the training data is never lost
- can be computationally expensive to label new instances
- finding appropriate distance metric can be difficult
- irrelevant attributes can have a negative impact

© University of Auckland    www.cs.auckland.ac.nz/~ian/    ian@cs.auckland.ac.nz

---

---

---

---

---

---

---

---