UNIVERSITY OF BERGEN

INFO 284 - Machine Learning

Introduction to Machine Learning

Bjørnar Tessem

UNIVERSITY OF BERGEN

Machine Learning

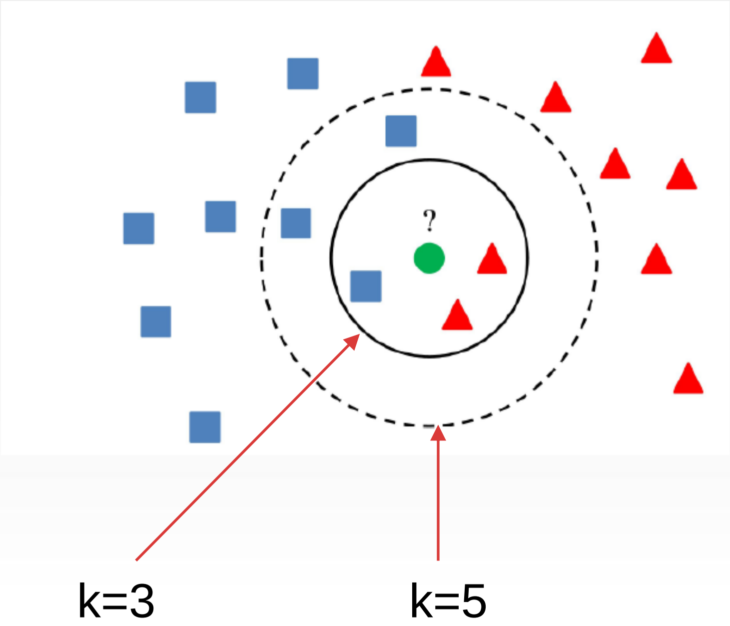
- Part of Artificial Intelligence
- Using data to get knowledge
  - Induction
  - Statistical approach
- Resulting MODELs are used for support of decision making in new situations

Types of machine learning

- Supervised learning
- Unsupervised learning
- Reinforcement learning

- Inductive logic programming
- Explanation based learning

K-nearest neighbours (kNN)



What is the type of the green dot?
- Red triangle or blue square

Principle:
- Use nearest neighbours and count each category
- Classify as the same as the majority

Example

Age	No.Cars	Owns house	No. children	Marital status	Owns a dog	Bought a boat	
66	**1**	**yes**	**2**	**widowed**	**no**	**yes**	5
52	**2**	**yes**	**3**	**married**	**no**	**yes**	2
22	0	no	0	married	yes	no	3
25	1	no	1	single	no	no	4
44	0	no	2	divorced	yes	no	4
39	1	yes	2	married	yes	no	5
26	1	no	2	single	no	no	4
40	3	yes	1	married	yes	no	5
53	**2**	**yes**	**2**	**divorced**	**no**	**yes**	5
64	2	yes	3	divorced	no	no	4
58	**2**	**yes**	**2**	**married**	**yes**	**yes**	5
33	1	no	1	single	no	no	4

X: 52 years, 0 cars, no house, 3 children, married, no dog

What will X do?

k-Nearest Neighbour

- The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.
- k is a user defined constant.
- A new data ‘point’ is assigned the label of the most frequent among the k ‘closest’ training data points
- ‘Closest’ or ‘most similar’ is defined in different ways

Supervised learning

- Given: a set of input-output pairs
- Learned: predicting output for a given input

input output

- Supervised learning involves observing several examples of a arbitrary $\boxed{\text{vector } \mathbf{x}}$ and an associated value or $\boxed{\text{vector } \mathbf{y}}$, then learning to predict **y** from **x**:

- by estimating $F(\mathbf{x}) = \mathbf{y}$

- To understand the above a quick dive into linear algebra

Linear algebra

- Algebra - the study of (mathematical) structures and the rules for manipulating these structures
- Linear - special types of structures

$$a_1 x_1 + \cdots + a_n x_n = b$$

- Scalar - single number
- Vector - array of numbers.
- An **array** is a container object that holds a fixed number of values of a single type.

Vector

- In a vector the numbers are arranged in an order.
- We can identify each number by its index.
- Vector notation - bold small case. Eg. **x**

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Matrix

- Matrix is a two dimensional array of numbers.
- We can identify each number in a matrix by **two** indices.
- Matrix notation - bold uppers case. Eg. **A**

$$\mathbf{A} = \begin{bmatrix} A_{1,1}, A_{1,2}, \cdots , A_{1_m} \\ A_{2,1}, \boxed{A_{2,2},} \cdots , A_{2_m} \\ \vdots \\ A_{n,1}, A_{n,2}, \cdots , A_{n_m} \end{bmatrix}$$

Tensor

- Tensor is an array of numbers arranged on a regular grid with a variable number of axes.
- An *n*-ranked tensor has *n* indices.
- Usage:
  - Sometimes used to describe a matrix of values together with how those values are transformed by some function
  - Sometimes used to represent whole collections of two-dimensional data

Tensor

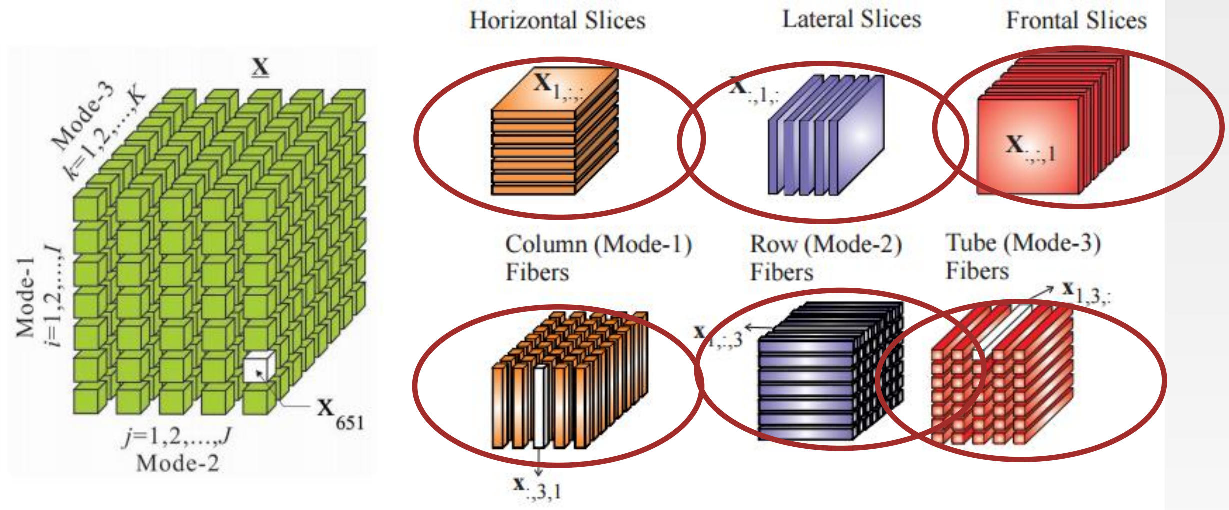


Figure 2: A 3rd-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$, with entries $x_{i,j,k} =$

kNN - Closest with continuous features

- Most often the **Euclidean distance** function is used
- Given two vectors (data points with **n** features)

$$\mathbf{x} = (x_1, x_2, \ldots, x_n) \qquad \mathbf{x}' = (x'_1, x'_2, \ldots, x'_n)$$

$$d(\mathbf{x}, \mathbf{x}') = \sqrt{(x'_1 - x_1)^2 + \boxed{(x'_2 - x_2)}^2 + \cdots (x'_n - x_n)^2}$$

$\mathbf{x}$ [4.7 3.2 1.3 0.2]

$\mathbf{x}'$ [4.6 3.1 1.5 0.2]

$$d(\mathbf{x}, \mathbf{x}') = \sqrt{(4.6 - 4.7)^2 + (3.1 - 3.2)^2 + (1.5 - 1.3)^2 + (0.2 - 0.2)^2)} = 0.245$$

kNN – Closest with discrete features

- Most often the **Hamming distance** function is used
- Given two vectors (data points with **n** features)

$$\mathbf{x} = (x_1, x_2, \ldots, x_n)$$
$$\mathbf{x}' = (x'_1, x'_2, \ldots, x'_n)$$

$$d_H(\mathbf{x}, \mathbf{x}') = |\{i \mid x_i \neq x'_i\}|$$

$\boxed{39}$	$\boxed{1}$	yes	$\boxed{2}$	married	yes	no
$\boxed{40}$	$\boxed{3}$	yes	$\boxed{1}$	married	yes	no

Some issues

- When k>1 neighbour
  - For discrete output values (classification) the majority category from k closest is selected
  - For continuous output values (regression) the average of the k closest is selected
- Higher number of neighbours and features → higher complexity, less quality
- Different variation in dimensions
  - Min-max-normalisation
- Combining continuous and discrete features
  - Euclid with Hamming distance for discrete features, real numbers for continuous features
- Case.based learning
  - Weighting features
  - Transform discrete features to indicator variables – one-hot encoding

Supervised learning

- Supervised learning involves observing several examples of an arbitrary vector **x** in the a feature space, together with an associated value or vector **y**, then learning to predict **y** from **x**.

- Given a training set of N example input-output pairs $(x_1,y_1), (x_2,y_2),\ldots,(x_N,y_N)$ where each y_i was generated by an unknown function $y_i=f(x_i)$, discover a function h that approximates the true function f.

hypothesis

Hypothesis space H

- Hypothesis space is the set of all functions *h* that approximate *f*.
- A learning problem is **realisable** if the hypothesis space contains the true function *f*.
  - We cannot always tell if a learning problem is realisable
- We go for approximations!

Finding a good hypothesis

- Learning is a search through the space of possible hypothesis to find one that **performs well**.

- We may measure how well a hypotheses performs in terms of **accuracy**. To do this we “give it” a **test set** of examples that are distinct from the training set.

- Accuracy - the fraction of examples from the test set for which the output was assigned correctly.

uib.no