Machine Learning Overview

Prof. Xiaowei Huang

https://cgi.csc.liv.ac.uk/~xiaowei/ai.html

(Attendance Code: **445048**)

In the last lecture,

- Module Information
- Contents of the module

Today's Content

Contents of the module (cont.)

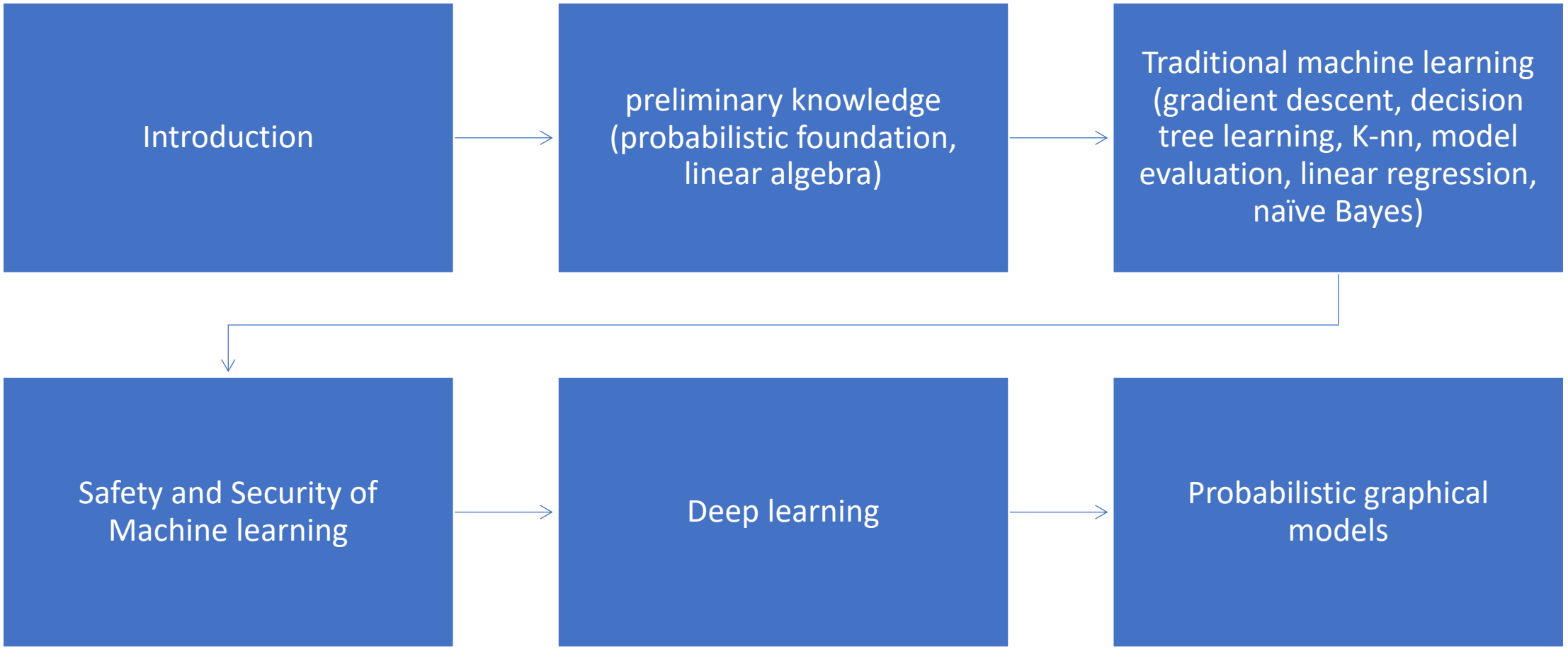
What is machine learning?

A few applications of machine learning

consider how to represent instances as fixed-length feature vectors

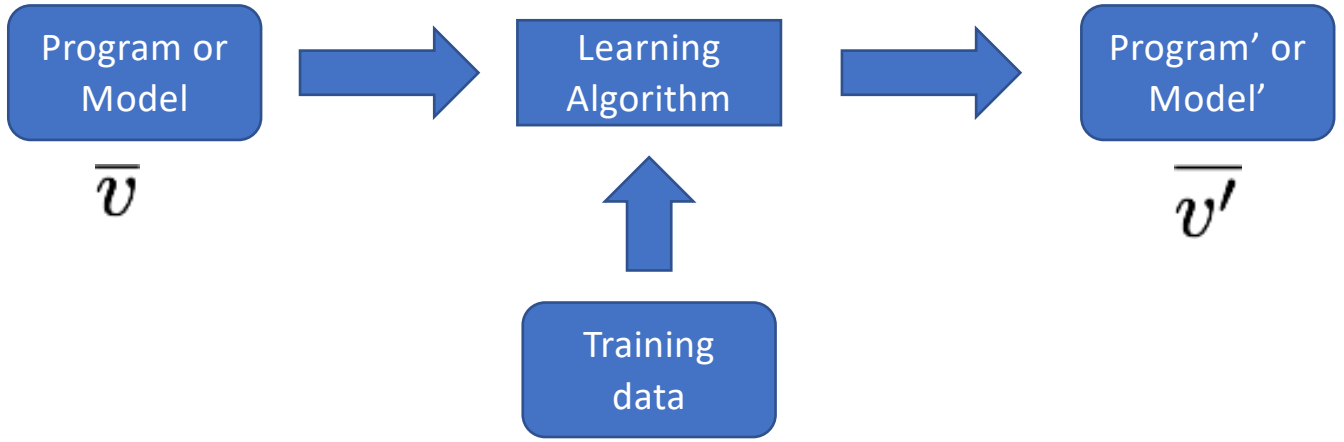
Data preprocessing

Contents of this module



What is Machine Learning?

- (Software) programs that can improve their performance by applying learning algorithm on training data
- Typically the program has a (large) number of parameters whose values are learnt from the data



Why learn instead of program?

- If the design of the agent can be improved, why wouldn't the designers just program in that improvement to begin with?
  - The designer cannot anticipate all possible situations that the agent might find itself in
  - The designers cannot anticipate all changes over time
  - Sometimes a human programmer has no idea how to program a solution

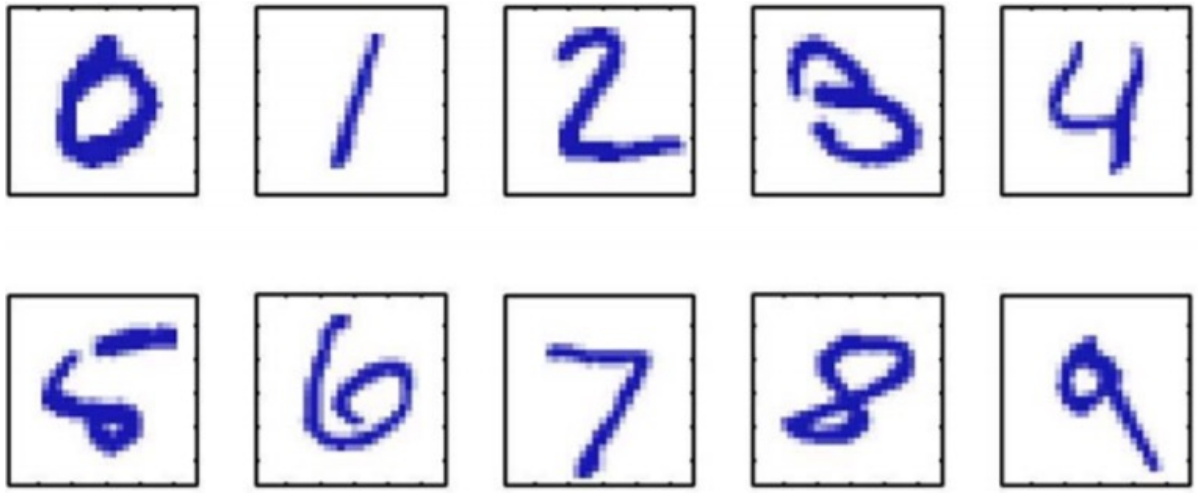
A few applications of machine learning

Where Machine Learning is used/useful?

- Can be applied in situations where it is very challenging (= impossible) to define rules by hand, e.g.:
  - Face detection
  - Speech recognition
  - Stock prediction

- When the application is able to be programmed with reasonable efforts, DO NOT use machine learning!

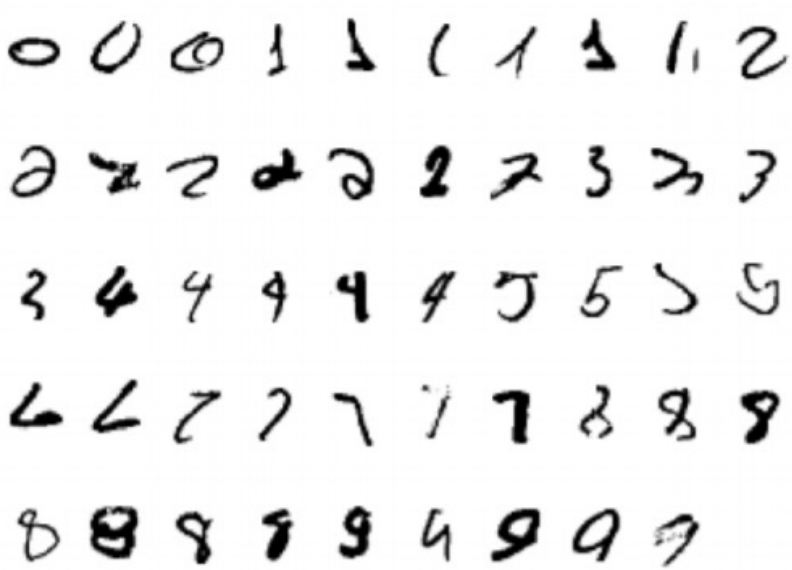
Example 1: hand-written digit recognition



Images are 28 x 28 pixels

Represent input image as a vector $x \in \mathrm{R}^{784}$,
learn a classifier f(x) such that

$$f : \mathrm{R}^{784} \rightarrow \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

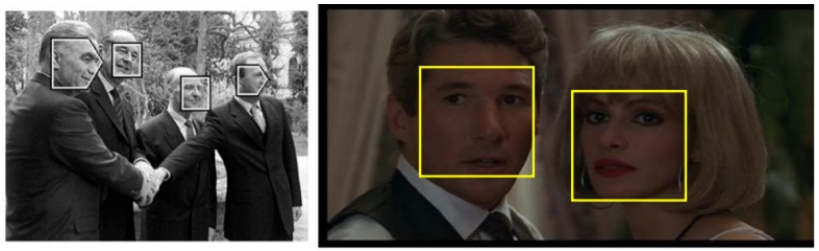


How to proceed …

- As a supervised classification problem
- Start with training data, e.g. 6000 examples of each digit

- Can achieve testing error of 0.4%
- One of the first commercial and widely used ML systems (for zip codes & checks)

Example 2: Face detection

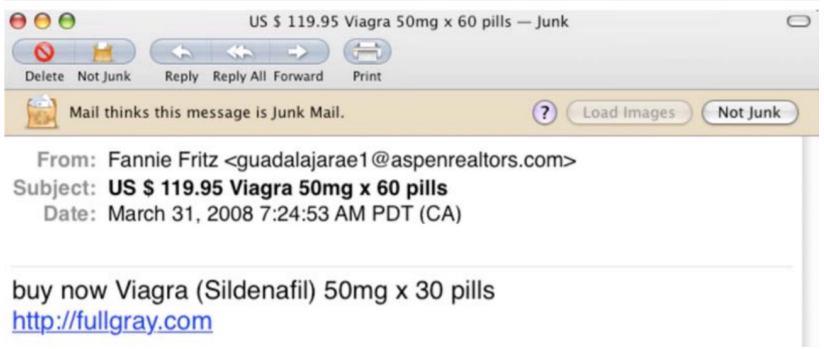


- Again, a supervised classification problem
- Need to classify an image window into three classes:
  - non-face
  - frontal-face
  - profile-face

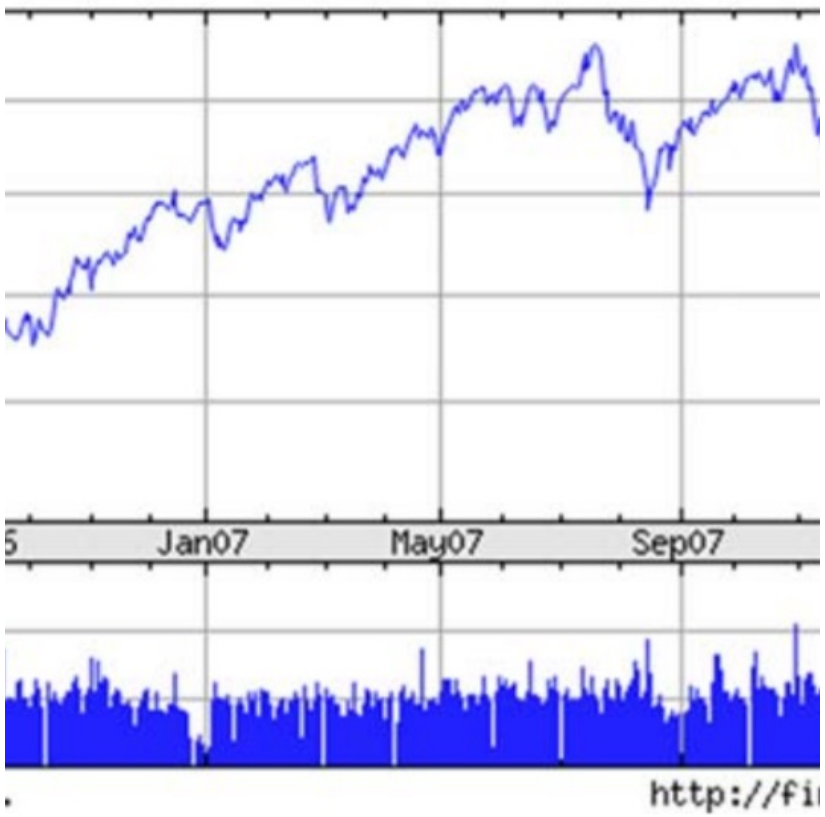
Classifier is learnt from labelled data

- Training data for frontal faces
  - 5000 faces
    - All near frontal
    - Age, race, gender, lighting
  - 10^8 non faces
  - faces are normalized
    - scale, translation (a **translation** is a geometric **transformation** that moves every point of a figure or a space by the same distance in a given direction)

Example 3: Spam detection



- This is a classification problem
- Task is to classify email into spam/non-spam
- Data x_i is word count, e.g. of viagra, outperform, “you may be surprised to be contacted” …
- Requires a learning system as “enemy” keeps innovating



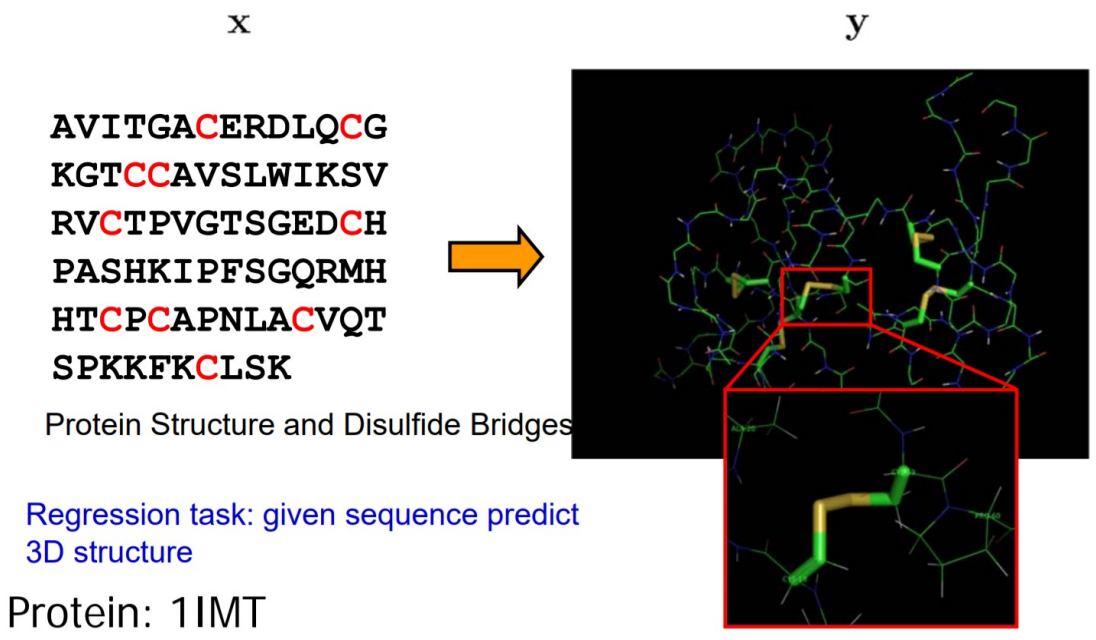
Example 4: Stock price prediction

- Task is to predict stock price at future date
- This is a regression task, as the output is continuous

Example 5: Computational biology

- **Protein structure prediction** is the inference of the three-dimensional structure of a protein from its amino acid sequence

- based on the dataset alone, the algorithm can learn how to combine multiple features of the input data into a more abstract set of features from which to conduct further learning



Web examples: Machine translation

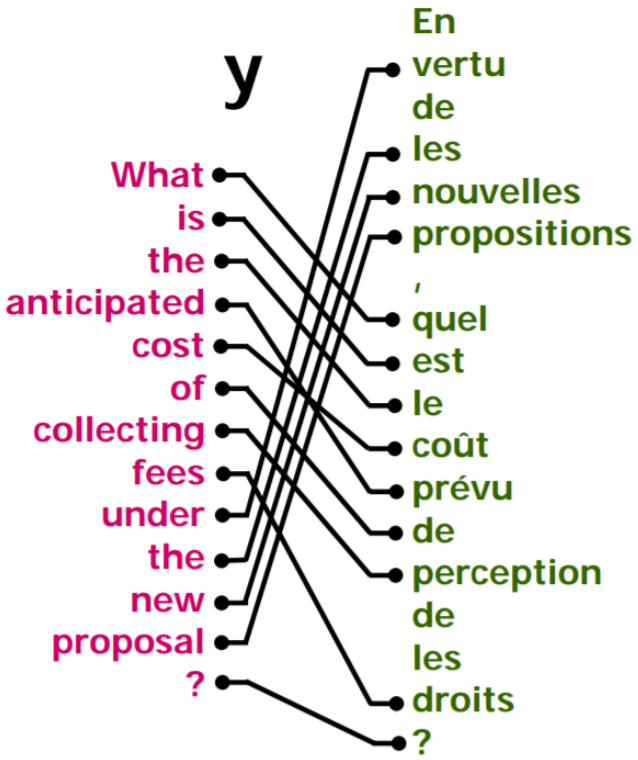
Use of aligned text

x

What is the anticipated cost of collecting fees under the new proposal?

En vertu des nouvelles propositions, quel est le coût prévu de perception des droits?

y



e.g. Google translate

Web examples: Recommender systems

People who bought Hastie …

Frequently Bought Together

Customers buy this book with Pattern Recognition and Machine Learning (Information Science and Statistics) (Information Science and Statistics) by Christopher M. Bishop

Price For Both: £104.95

Add both to Basket

Customers Who Bought This Item Also Bought

Page 1

Pattern Recognition and Machine Learning (Infor...
by Christopher M. Bishop
★★★★☆ (4) £48.96

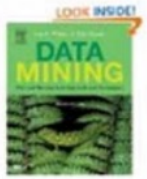
Show related items

MACHINE LEARNING (Mcgraw-Hill International Edit) by Thom M. Mitchell
★★★★★ (3) £42.74

Show related items

Pattern Classification, Second Edition: 1 (A Wi...
by Richard O. Duda
★★★★★ (1) £78.38

Show related items



Data Mining: Practical Machine Learning Tools a...
by Ian H. Witten
★★★★★ (1) £37.04

Show related items

represent instances as fixed-length feature vectors

Can I eat this mushroom?

- I don’t know what type it is – I’ve never seen it before. Is it edible or poisonous?

Can I eat this mushroom?

suppose we're given examples of edible and poisonous mushrooms (we'll refer to these as *training examples* or *training instances*)

edible

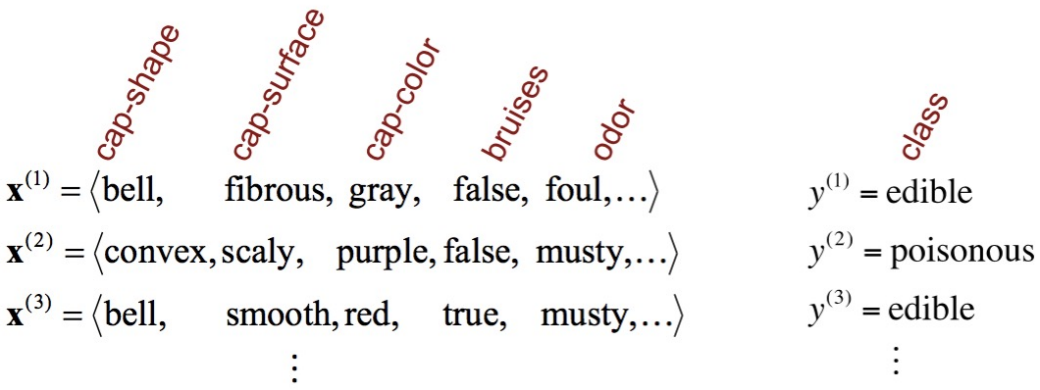



poisonous



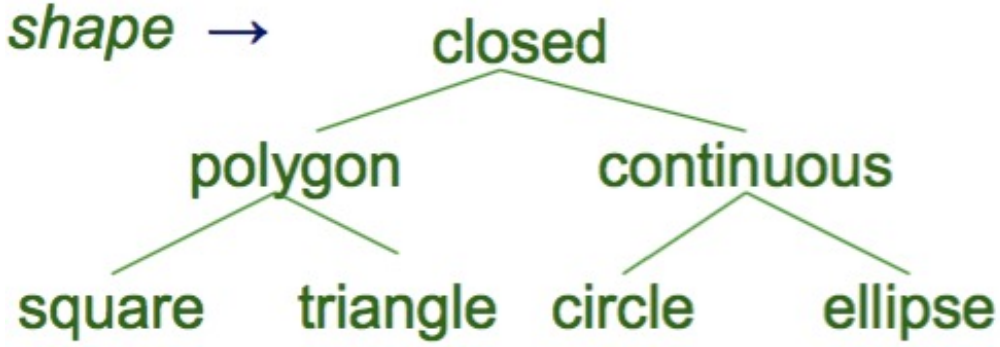

Representing instances using feature vectors

- we need some way to represent each instance
- one common way to do this: use a fixed-length vector to represent *features* (a.k.a. *attributes*) of each instance
- also represent *class label* of each instance



Standard feature types

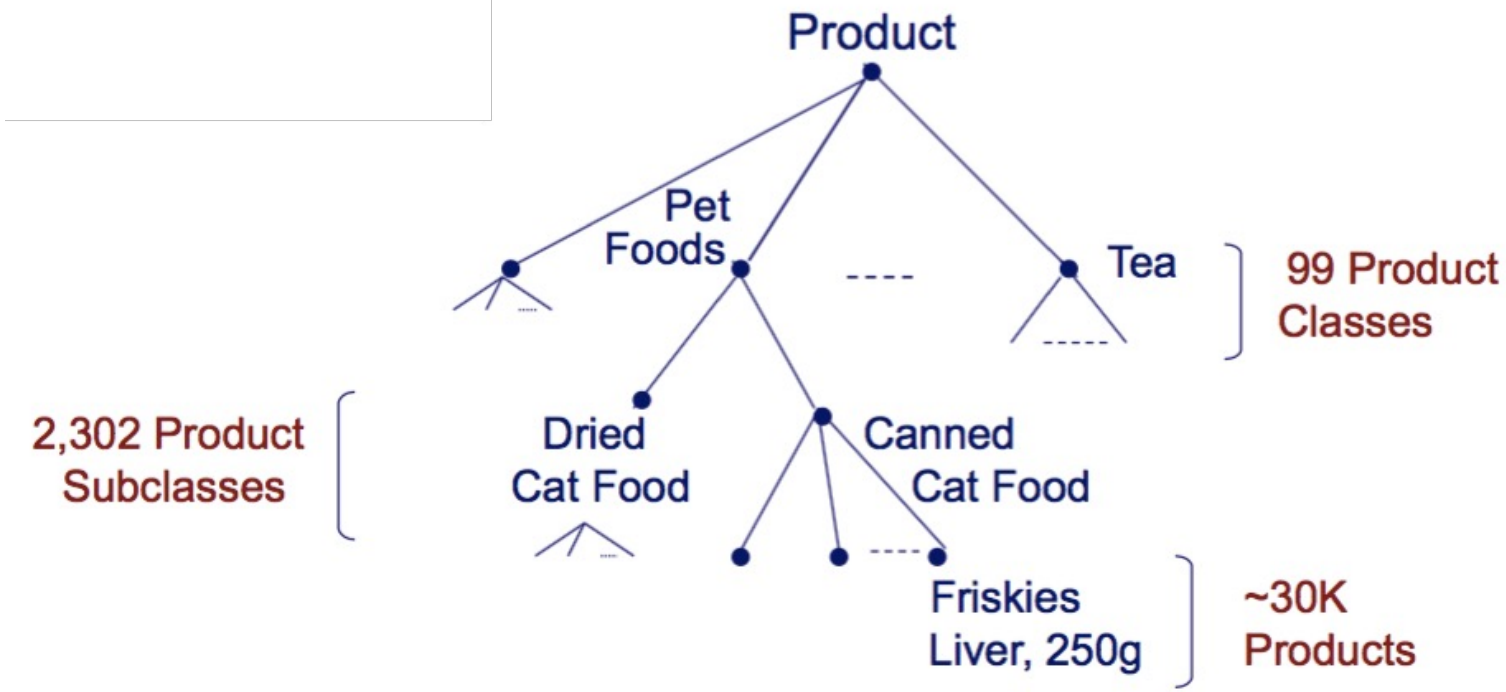
- *nominal* (including Boolean)
  - no ordering among possible values
    - e.g. *color* ∈ {*red, blue, green*} (vs. *color = 1000* Hertz)
- *ordinal*
  - possible values of the feature are totally ordered e.g. *size* ∈ {*small, medium, large*}
- *numeric (continuous)*
  - *E.g., weight* ∈ [0...500]
- *hierarchical*
  – possible values are partially *ordered* in a hierarchy



Feature hierarchy example

- Lawrence et al., *Data Mining and Knowledge Discovery* 5(1-2), 2001

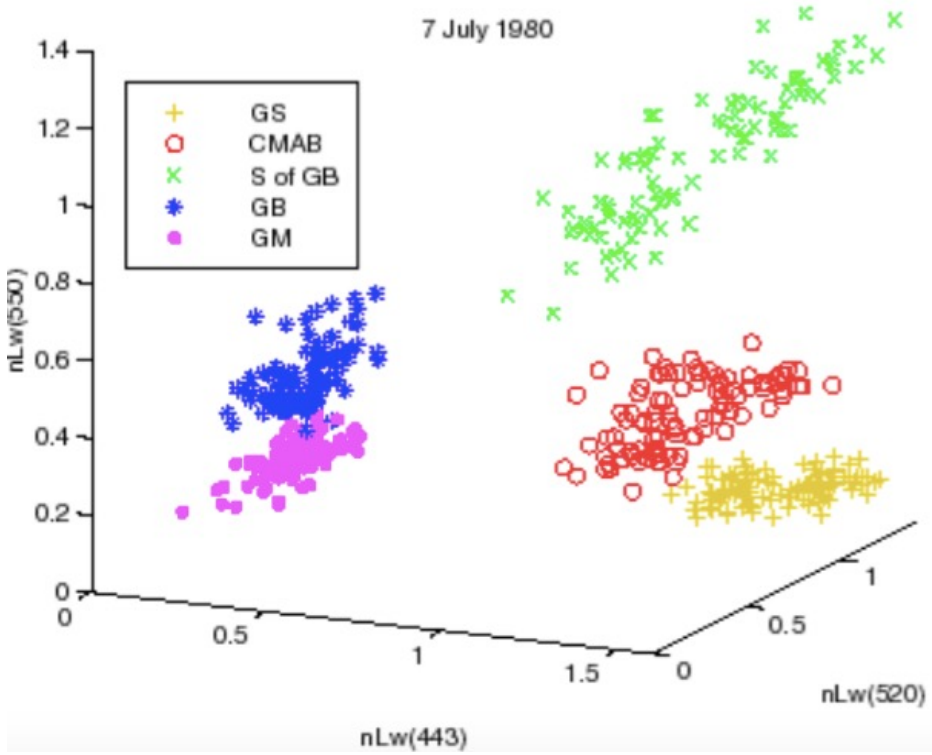
Structure of one feature



Feature space

- we can think of each instance as representing a point in a d-dimensional feature space where d is the number of features

example: optical properties of oceans in three spectral bands [Traykovski and Sosik, *Ocean Optics XIV Conference Proceedings*, 1998]



How about a 3-dimensional space for height, weight, and body fat percentage?

Another view of the feature-vector representation: a single database table

	feature 1	feature 2	. . .	feature d	class
instance 1	0.0	small		red	true
instance 2	9.3	medium		red	false
instance 3	8.2	small		blue	false
. . .					
instance n	5.7	medium		green	true

Data Preprocessing

10.4.3.1 Mean Normalization

removes the mean from each data sample, i.e.,

$$\mathbf{x}' = \mathbf{x} - \bar{\mathbf{x}} \tag{10.43}$$

where $\bar{\mathbf{x}} = \frac{1}{|D|} \sum_{\mathbf{x} \in D} \mathbf{x}$ is the mean of the dataset D.

10.4.3.2 Standardization or Normalization

requires, on top of the mean normalisation, all features to be on the same scale, i.e., every sample $\mathbf{x}$ is converted into

$$\mathbf{x}' = \frac{\mathbf{x} - \bar{\mathbf{x}}}{\sigma_D} \tag{10.44}$$

where σ_D is the standard deviation of the dataset D.

Data Preprocessing

10.4.3.3 Whitening

requires that the covariance matrix of the converted dataset is the identity matrix—1 in the diagonal and 0 for the other cells. It first applies the mean normalisation on the dataset D to get D', and then apply a whitening matrix W on every sample, i.e., let $\mathbf{x}'' = \mathbf{W}\mathbf{x}'$, such that $\mathbf{W}\mathbf{W}^T = \Sigma^{-1}$ and Σ is the non-singular covariance matrix of D'.

Depending on what $\mathbf{W}$ is, we have Mahalanobis or ZCA whitening ($\mathbf{W} = \Sigma^{-1/2}$), Cholesky whitening ($\mathbf{W} = \mathbf{L}^T$ for $\mathbf{L}$ the Cholesky decomposition of Σ^{-1}), or PCA whitening ($\mathbf{W}$ is the eigen-system of Σ^{-1}).