Machine learning

(521289S)

Prof. Tapio Seppänen
Center for Machine Vision and Signal Analysis
University of Oulu

Initial Lecture Contents

- First part:
  - General information about the course
- Second part:
  - Statistical pattern recognition basics

PART I

GENERAL INFORMATION

Artificial intelligence cources at ITEE

- Introduction to artificial intelligence
- Artificial intelligence
- Affective Computing
- Big data processing and applications
- Machine learning
- Machine vision
- Natural language and text mining
- Multi-modal data fusion
- Towards data mining

Machine learning (521289S, 5 cu)

- Responsible teacher
  - Prof. Tapio Seppänen

- Course focuses on **statistical pattern recognition**
  - A central part of artificial intelligence and machine learning

- After completing the course the student can
  1. Solve basic statistical calculation problems of pattern recognition
  2. Design simple optimal classifiers from the basic theory and assess their performance
  3. Implement and test a pattern recognition system.

Course Contents in Brief

- Introduction
- Bayesian decision theory
- Discriminant functions
- Parametric and non-parametric classification
- Feature extraction and selection
- Classifier design
- Example classifiers
- Neural networks such as Perceptron

Contents in More Detail

- Principles of statistical pattern recognition (tilastollinen hahmontunnistus)
  - Feature vector (piirrevektori), feature space (piirreavaruus), distributions (jakaumat), decision surface (päätöspinta), discriminant function (diskriminanttifunktio), distances (etäisyydet), feature selection (piirteen valinta), classifiers (luokittelijat), training and testing (opetus ja testaus)
- Bayesian decision theory (Bayesin päätösteoria)
  - Derivation of a classifier, classifiers based on Normal distribution
- Estimation of distributions: parametric and non-parametric estimation
  - MLE, kNN, Parzen
- Parametric and non-parametric classifiers
- Linear discriminant classifiers
  - Linearly separated classes, gradient search (basic, Perceptron), MSE, LMS
- Specificity, sensitivity, positive and negative predictive values, ROC
- Bayesian networks
- Neural networks
  - Structure, training and operation of a neuron and a neural network
  - Perceptron: properties, XOR problem, training algorithm
  - Multi-layer Perceptron: the basic principle

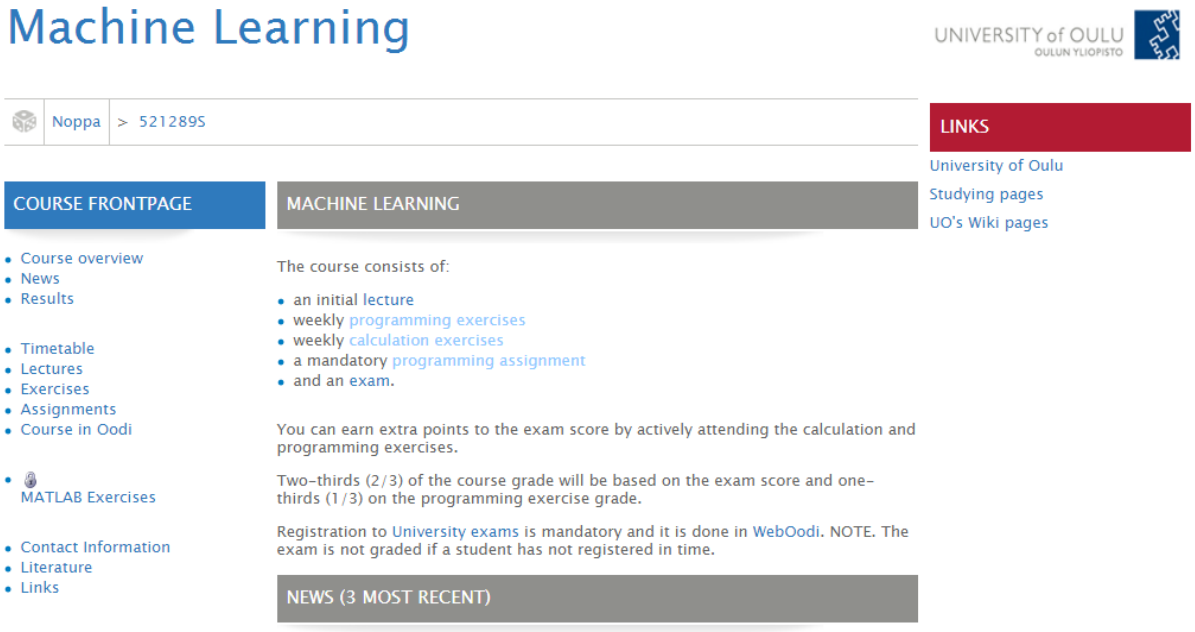
Course Material

- Duda RO, Hart PE, Stork DG, Pattern classification, John Wiley & Sons Inc., 2nd edition, 2001. Pages: 1-27, 29-46, 51-54, 56-62, 84-90, 161-172, 174-179, 182-184, 187-188, 215-233, 239-242, 245-246, 282-303, 306-318. See also the revised chapter available online: ftp://ftp.wiley.com/public/sci_tech_med/pattern/DHS2.11Revised.pdf
- Haykin S, Neural networks, MacMillan College Publishing Company, pp. 408-434 or 1999, pp. 465-499 (Self-Organizing Maps)
  - Note: Chapters 1-4 from Haykins (1999) book contain a well written representation of the topics related to Artificial Neural Networks and to Multilayer Perceptrons relevant to this course. The content is partially overlapping the one in Duda, Hart & Stork (2001), but explained perhaps more clearly in Haykins book (e.g., the section related to backpropagation algorithm).
- Selected articles:
  - Pudil PJ, Novovicova J, Kittler J (1994). Floating search methods in feature selection. Pattern Recognition letters 15 (11), 1119-1125
  - Freund Y, Shapire R (1999). A short introduction to boosting. Journal of Japanese Society for Artificial Intelligence, 14(5): 771-780
- Calculation exercises, programming exercises

Course Website @Noppa

- https://noppa.oulu.fi/noppa/kurssi/521289s

- Communication of important information is performed via the web pages. Check the News section periodically for any schedule changes, important events etc. !



Course Implementation

- The parts of the course:
  - Introductory lecture (this one)
  - Calculation exercises (earn extra points!)
  - Programming exercises (earn extra points!)
  - Programming assignment (compulsory!)
  - Examination (compulsory!)

- Remember to register to the course and exam in WebOodi: https://weboodi.oulu.fi/oodi

Calculation Exercises

- On Thursday at 12.15-14.00 in L8
- The exercises are designed to help understand the background theory for the programming exercises
- Earn max 3 extra points
  - Will be added to your exam result
  - Participate and work on the assigned problems
- The extra points cannot be earned remotely. Physical participation is required
- The material for the problems will be announced on the course website about a week before each exercise; study it already before the exercise
- The solutions will be published on the website on the next day after the exercise

- Responsible assistant: M.Sc. Antti Isosalo
- Other assistants: Mr. Ziaul Hoque

Programming Exercises

- On Tuesday at 12.15-14.00 in MA336 & MA337
- The programming exercises are designed to help performing well in the programming assignment
- Earn max 3 extra points
  - Will be added to your exam result
  - Participate and work on the assigned problems
- The extra points cannot be earned remotely. Physical participation is required
- The solutions will be published on the website
- **Important**: study MATLAB tutorial from the website before the first exercise!
  - ”Matlab Tutorials”
  - ”Matlab Getting Started Guide”

- Responsible assistant: Dr. Anja Keskinarkaus
- Other assistants: Mr. Brian Irvine, Mr. Ziaul Hoque

Programming assignment

- Designing MATLAB software for a pattern recognition application
  - Training and testing with given data
  - Many useful SW components are programmed in the programming exercises
- Each student will do the assignment solo
- About the assignment:
  - The software is first developed and tested independently with the given data
  - The software is uploaded to an automatic ranking server
  - The server executes the software with other (similar) data and yields a performance figure
  - Performance figures are placed on the website and ranked
- Detailed information will be given on the website
  - Instructions for the assignment
  - Returning the assignment for grading
- Plagiarism incidents will be handled according to university guidelines
- Responsible assistant: Mr. Kai Noponen

Course Schedule

- Calculation exercises on 10.1. – 28.2.
  - On Thursday at 12.15-14.00
  - Lecture room L8

- Programming exercises on 15.1. – 26.2.
  - On Tuesday at 12.15-14.00
  - Lecture room MA336 & MA337

- Programming assignment will be given on week 9 (25.2.-1.3.2019).
  - Deadline for returning assignments is 30.3.2019

- Exam on 18.3.2019. Retake on 9.5.2019.
  - Remember to register in WebOodi: https://weboodi.oulu.fi/oodi
  - Possible summer exam depends on the number on voting students

Examination

- Usually:
  - 2 questions from the theory (course book)
  - 2 questions from the calculation exercises
- Max 4x6=24 points

Course Grading

- Course grade
  - Examination: 1-5, weight 2/3
  - Programming assignment: 1-5, weight 1/3
  - Extra points for the exam:
    - Calculation exercises: 0-3 points (active participation)
    - Programming exercises: 0-3 points (active participation)
    - In total: 6 points = one exam question
  - Grade = 2/3*(exam plus extra points) + 1/3*(programming assignment)

- The grade will be input to the Oodi system when **both exam and the final programming assignment** have been accepted

Contact information

- Can be found on the course web page
- Prefer email!
  - No free visiting times; must be agreed beforehand
- Choose mail recipient based on the issue:
  - Course overall -> Tapio
  - MATLAB exercises -> Anja
  - Calculation exercises -> Antti
  - Programming assignment -> Kai

PART II

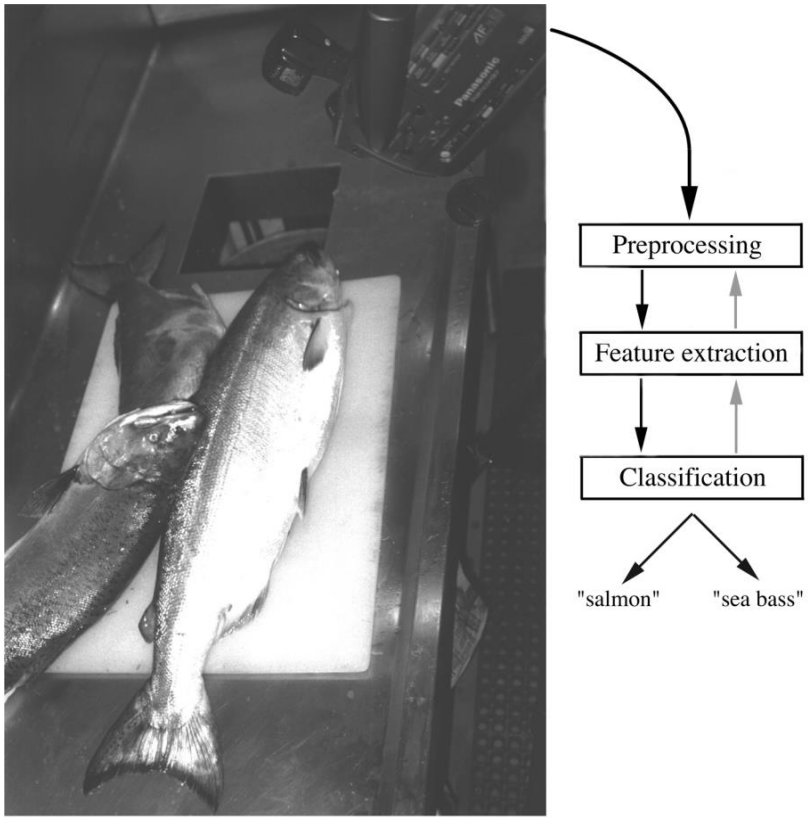
PATTERN RECOGNITION BASICS

Pattern Recognition?

- A branch of machine learning
- Focuses on methods and algorithms to recognize and discover patterns and regularities in data
- Statistical PR uses statistical methods
  - E.g. choose optimal methods in probabilistic sense
- Observations can come in many forms such as these signals and sensors:
  - Temperature (thermometer)
  - Humidity (hygrometer)
  - Sounds (microphone)
  - Images (camera)
  - Position (GPS, inertial navigation)
  - Speed/velocity (speedometer)
  - Acceleration (accelerometer)
  - Distance (ultrasound, laser, radar, …)
  - Touch (touchscreen)
  - Current
  - Voltage
- An essential part of any system that tries to *observe it's surroundings* and *choose meaningful actions* based on the observations
  - Automatic speech recognition, Speaker identification, Prosodic analysis, …
  - Computer vision (quality inspection, robotic vision, …)
  - Image analysis (fingerprint identification, identification from facial images, …)
  - Video surveillance (crossroads, taxi stations, stores & warehouses, …)
  - Spectral measurements (Remote sensing, plant vitality, …)
  - Context sensitive computing (device/program adapts based on the situation/state of the user)
  - Information retrieval systems (automated analysis of databases, …)
  - Biomedical system analysis (EEG, ECG, …)
  - Optical character recognition (OCR), Natural language processing (NLP), …

PR System Toy Example: Fish species classification

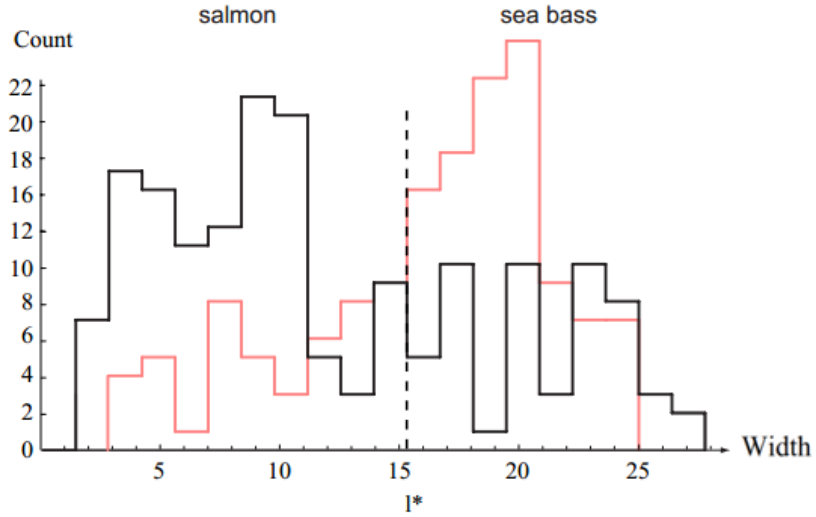
A fish packing plant wants to automate the process of sorting incoming fish on a conveyor belt according to species.



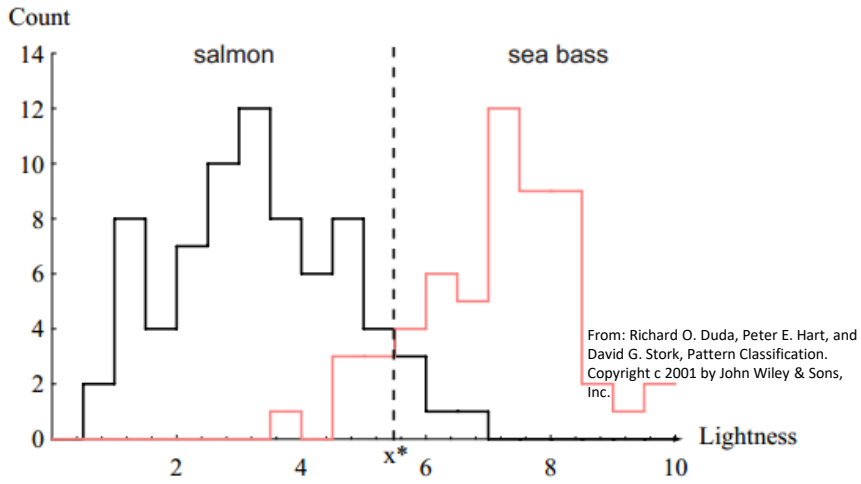
From: Richard O. Duda, Peter E. Hart, and David G. Stork, Pattern Classification. Copyright c 2001 by John Wiley & Sons, Inc.

- The objects to be classified: Fish
- Pattern recognition steps
  - Sensed by a transducer: camera
  - Images are preprocessed
  - Features extracted
  - Classification emitted
    - either "salmon" or "sea bass"
- Information flow is often chosen to be from the data source to the classifier ("bottom-up") as here
- Some systems employ "top-down" flow as well, in which earlier levels of processing can be controlled based on the tentative or preliminary response in later levels (gray arrows).
- Yet others combine two or more stages into a unified step, such as simultaneous segmentation and feature extraction.

Classification Based on Features



Histograms for the **width** feature for the two categories.



Histograms for the **lightness** feature for the two categories.

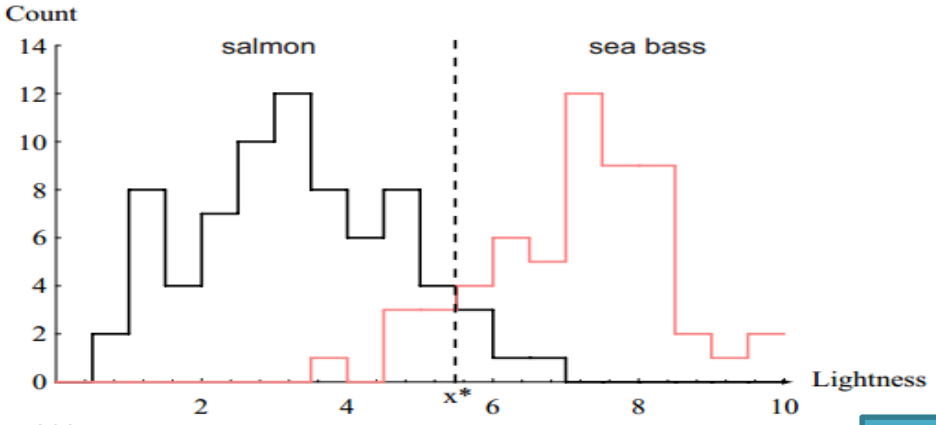
- No single threshold value l^* or x^* (decision boundary) will serve to unambiguously discriminate between the two categories
- Using either width or lightness alone, we will have some errors
- The values l^* & x^* lead to the smallest number of errors on average

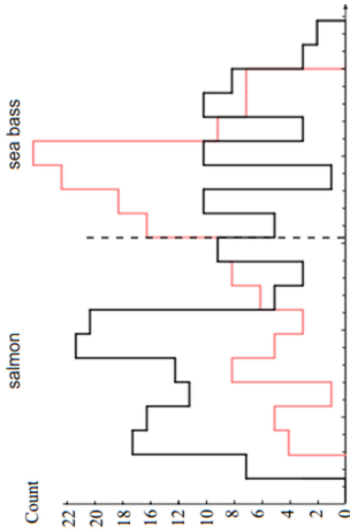
Feature vectors & Feature space

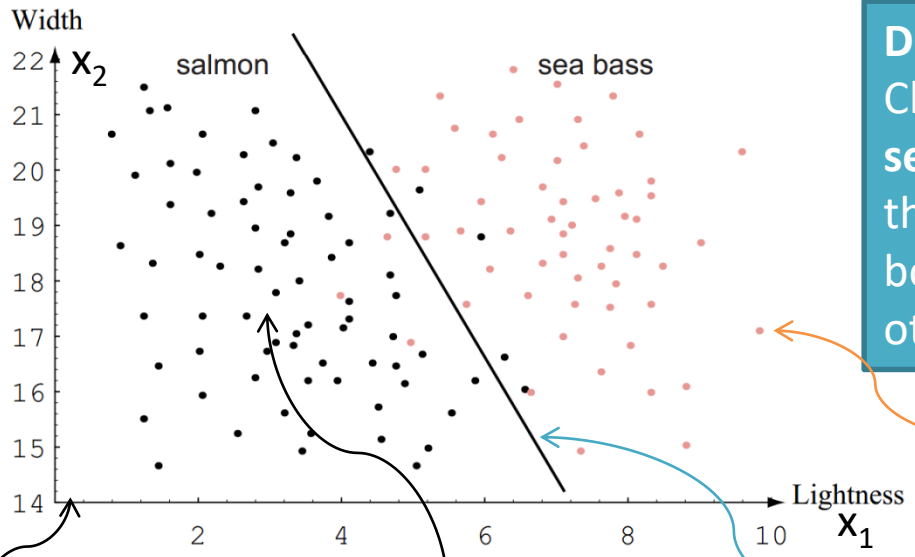
Let **lightness** = x_1
and **width** = x_2

Combine to form
a feature vector

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







Decision rule:
Classify sample as **sea bass** if **x** is above the decision boundary; otherwise as salmon

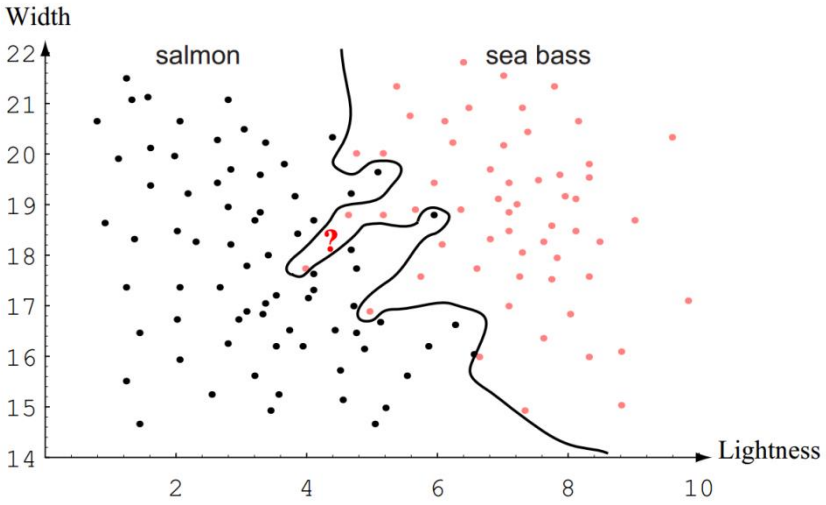
Sample (sea bass)

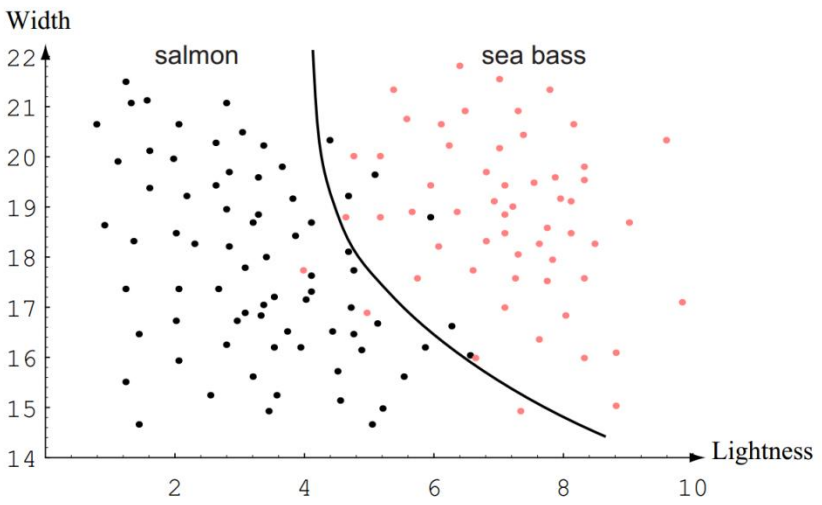
Feature space

Cluster (salmon)

Decision boundary

Complexity and Overlearning

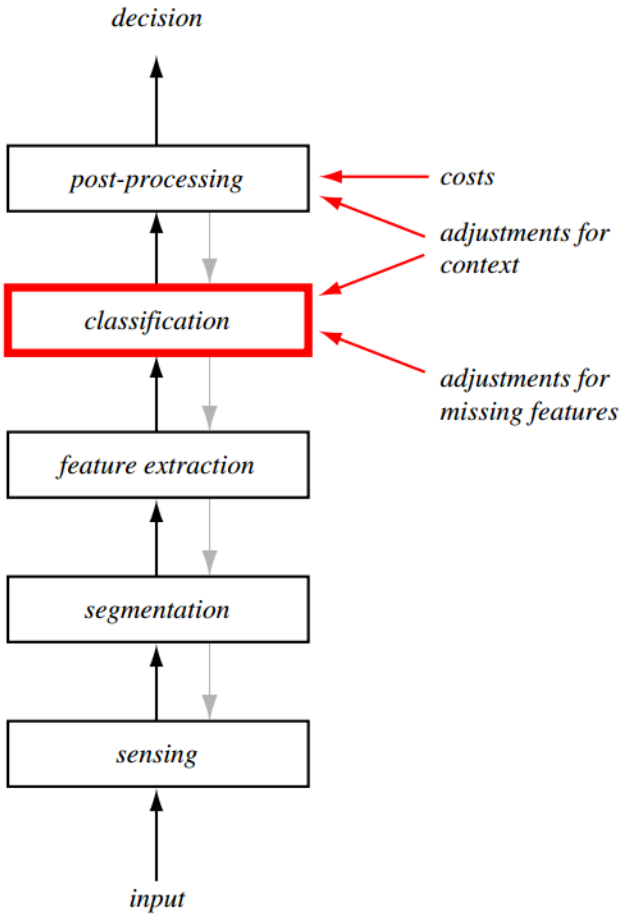




- Overly complex models for the fish will lead to decision boundaries that are complicated.
- While such a decision may lead to perfect classification of our training samples, it could lead to poor performance on new patterns:
  - The novel test point marked ? is evidently most likely a salmon, whereas the complex decision boundary shown leads it to be misclassified as a sea bass.

- The decision boundary shown here might represent the optimal tradeoff between performance on the training set and simplicity of classifier.

General Structure of a PR System



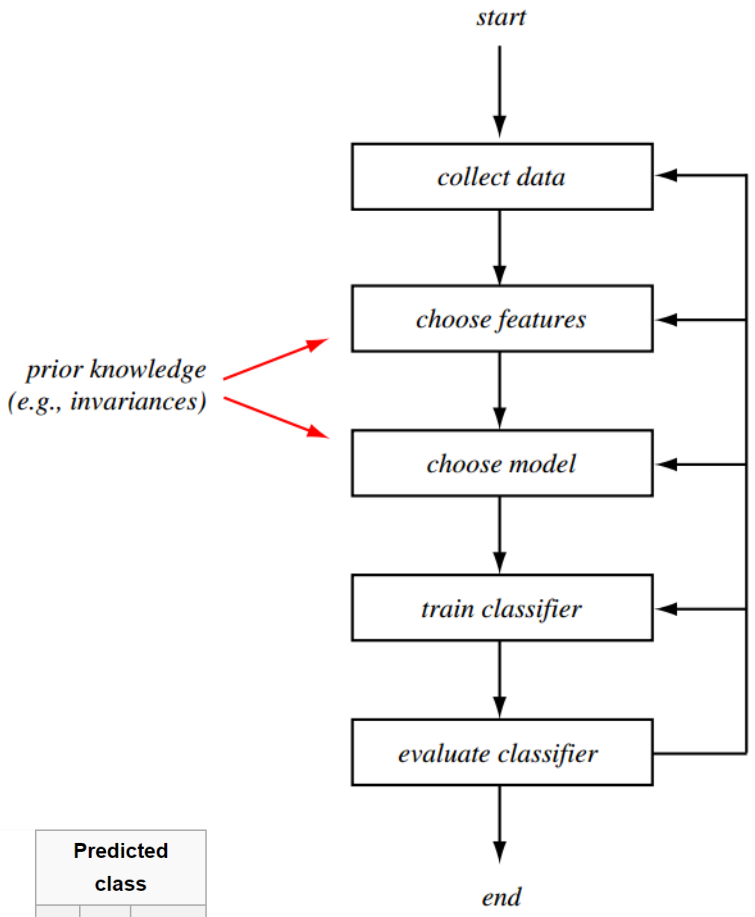
Many PR systems can be partitioned into several components

- **Sensor**
  - Converts images or sounds or other physical inputs into signal data
- **Segmentor**
  - Isolates sensed objects from the background or from other objects
- **Feature extractor**
  - Measures object properties that are useful for classification
- **Classifier**
  - Uses these features to assign the sensed object to a category
- **Post processor**
  - Decides on the appropriate action
  - Can take into account other considerations,
    - the effects of context
    - the costs of errors
    - …

Designing PR Systems

Typical steps in a design cycle

- Data collection
  - To train the system
  - To test the system
- Observed data characteristics have an impact on the workflow
  - Choice of discriminating features
  - Choice of models for the different categories
- Training process
  - Use some or all of the data to determine the system parameters
  - Learning data, testing data (+validation data)
    - hold-out, leave-k-out, bootstrapping…
- Evaluate results
  - The results may call for repetition of various steps in this process in order to obtain satisfactory outcomes



Confusion matrix

		Predicted class		
		Cat	Dog	Rabbit
Actual class	Cat	5	3	0
	Dog	2	3	1
	Rabbit	0	2	11

Learning Variants

- Supervised learning
  - Training data samples have a class label
  - Classifier uses labelled data to learn decision boundaries / surfaces
  - “The typical way”

- Unsupervised learning
  - Class labels are not supplied in the learning phase
  - Good for statistical clustering analysis
    - Learning the “natural groupings” of data

- Reinforcement learning
  - Classifier is only told whether it succeeded or not
  - Analogous to a critic who merely states that something is right or wrong, but does not say specifically how it is wrong

- Active learning, semi-supervised learning, deep learning,…