DESIGN OF CLASSIFIERS

Dr. Umarani Jayaraman
Assistant Professor

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,
DESIGN AND MANUFACTURING,
KANCHEEPURAM

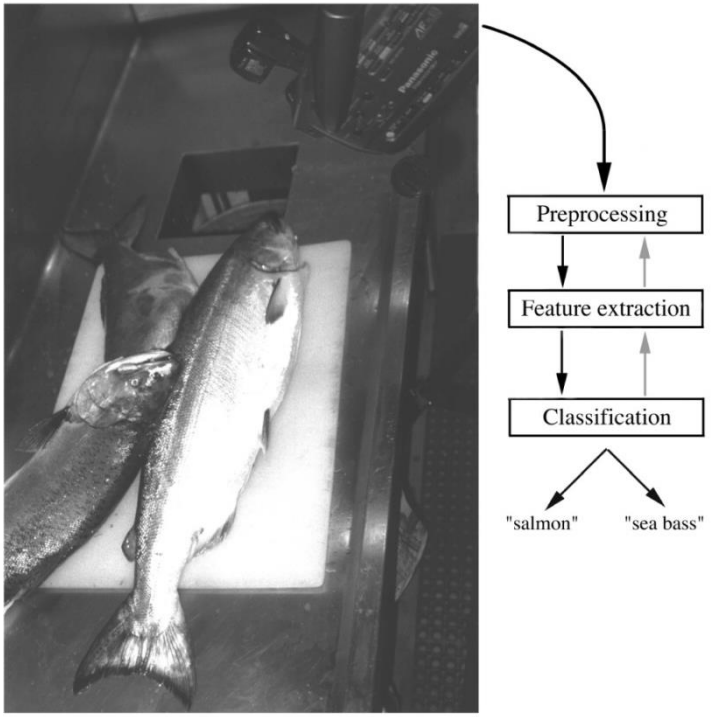
A simple PR problem

Problem: Sorting incoming fish on a conveyor belt.

Assumption: Two kind of fish:

(1) sea bass

(2) salmon



Challenges of PR- Feature Extraction

- Correlated features / Redundant Features
- Large number of features /Curse of dimensionality
- Noise in measurements/ Noisy or Missing Features
- Large variation within class: A lot of variability in patterns (feature vector) of a single class, the intra-class similarity is low
- Feature vectors of patterns from different classes can be arbitrarily close, and inter-class similarity is high
- Given this much variability, it is not so easy to design the classifier

Problem of classification

- Classification: the process of assigning an object to one of the known classes (true class)
- It does the task by **Learning**
- Ideally, a classifier should have
  - **Good generalization**

What do we have?

- The only information is available for the design is – A training set of sample/example pattern
- Training set: (X_i , y_i)
- Here X_i is a sample of class y_i

How do we generate the training set?

- Generation of the training set- take representative patterns of the known category called **data collection**
- Obtain the feature vectors from those data (choice of feature measurements)
- Now **learn** an appropriate function *h* as the classifier
- Test and validate the classifier on more/unseen data

A simple PR problem

- **Problem:** Sorting incoming fish on a conveyor belt.
- **Assumption:** Two kind of fish:
- (1) sea bass
- (2) salmon

Design of classifier:

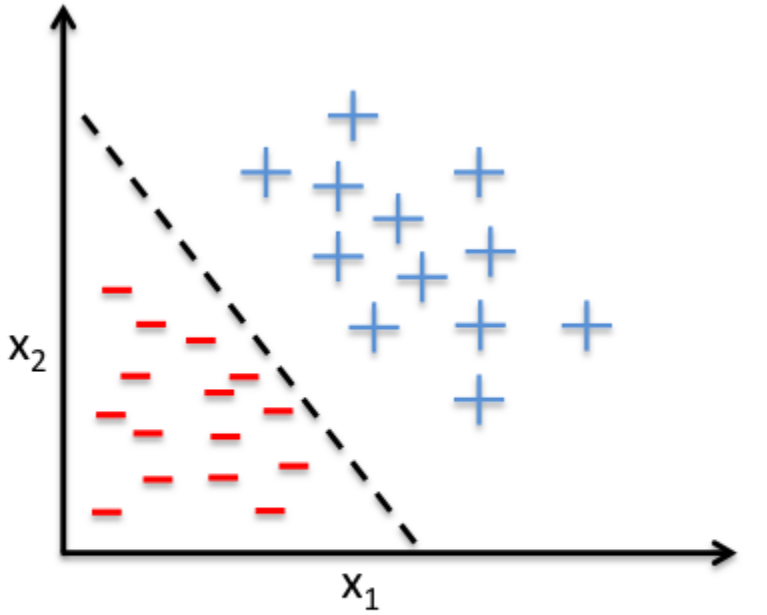
$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

- x_1: length
- x_2: lightness

- A classifier ax_1+bx_2
- If $ax_1+bx_2 > T$ then sea bass, otherwise it is salmon
- a,b,T are known as learning parameters
- What values to use for a,b,T ?

Design of classifier:

- The classifier is linear here
  - x_1: length
  - x_2: lightness

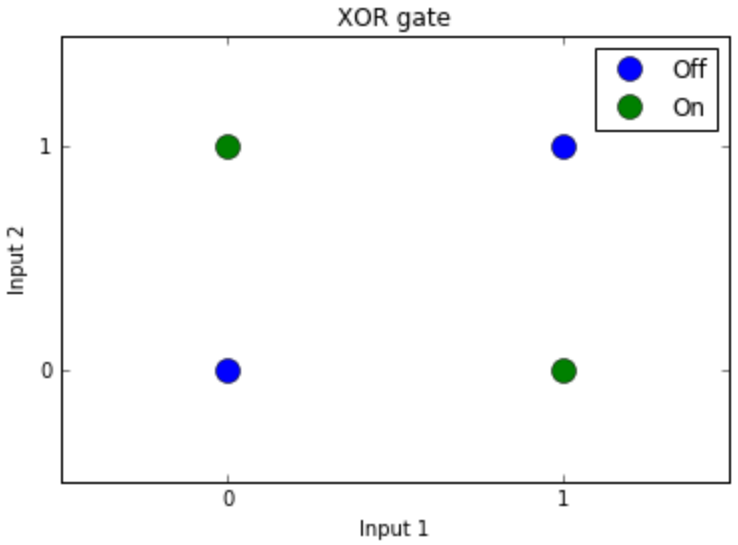


Example of a linear decision boundary for binary classification.

Another example

- XOR Gate

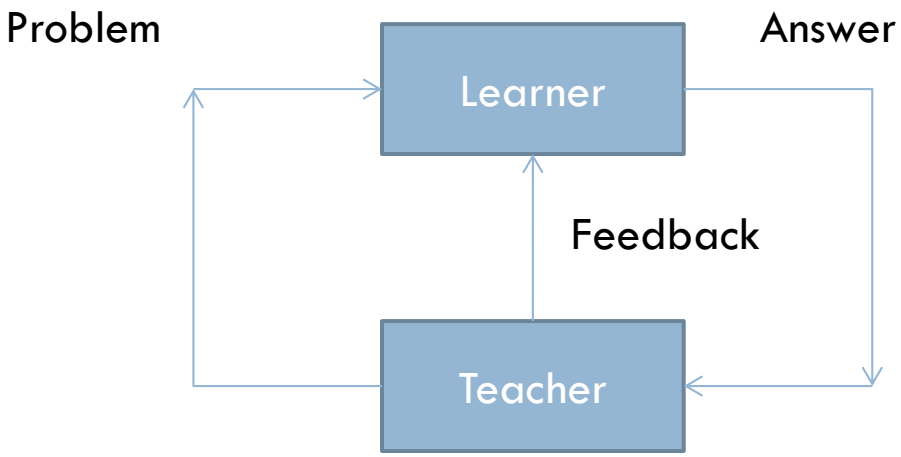
x_1	x_2	x_1 XOR x_2
0	0	0
0	1	1
1	0	1
1	1	0



- The classifier is non-linear here

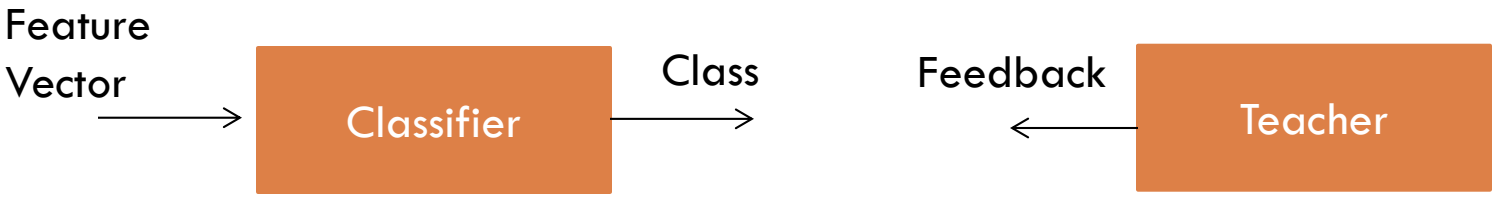
Learning from samples/examples

- Designing a classifier is a typical problem of learning from samples/ examples
- Also called learning with a teacher
- Nature of feedback from a teacher can be different in the context of PR



Learning from examples

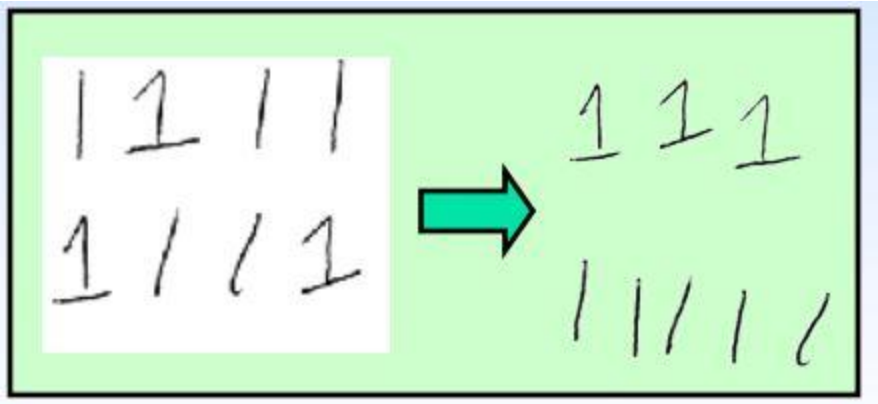
- In the context of PR



- Supervised Learning: The teacher gives the true class label for each feature vector
- Unsupervised Learning: No teacher input (clustering problem)
- Reinforcement Learning: Noisy assessment of performance (eg: correct/incorrect)

Learning from examples

- Unsupervised Learning: No teacher input (clustering problem)
- Clustering (learning categories)

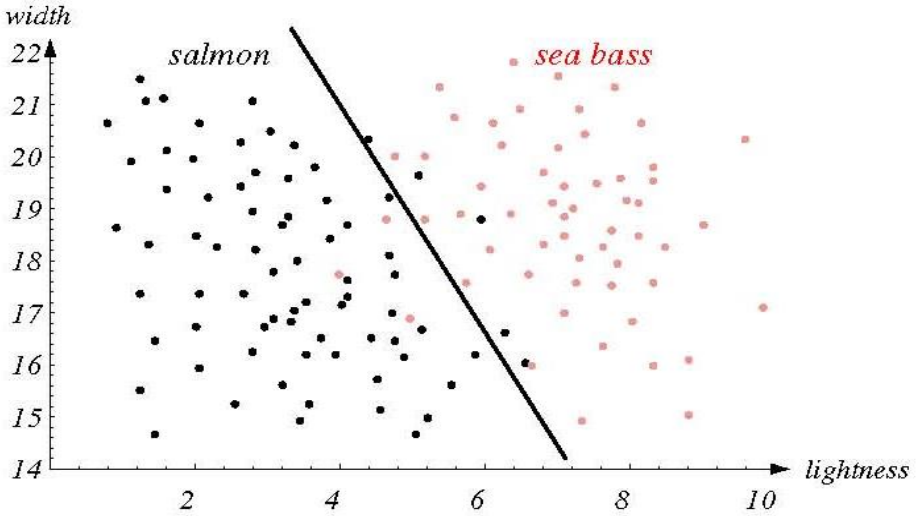


Learning from examples

- When the class labels of training patterns as given by teacher are noisy, we consider it as supervised learning with noise or classification noise
- We design a classifier with training samples (learning from examples)
- Many classifier algorithms do supervised learning

Design of classifier

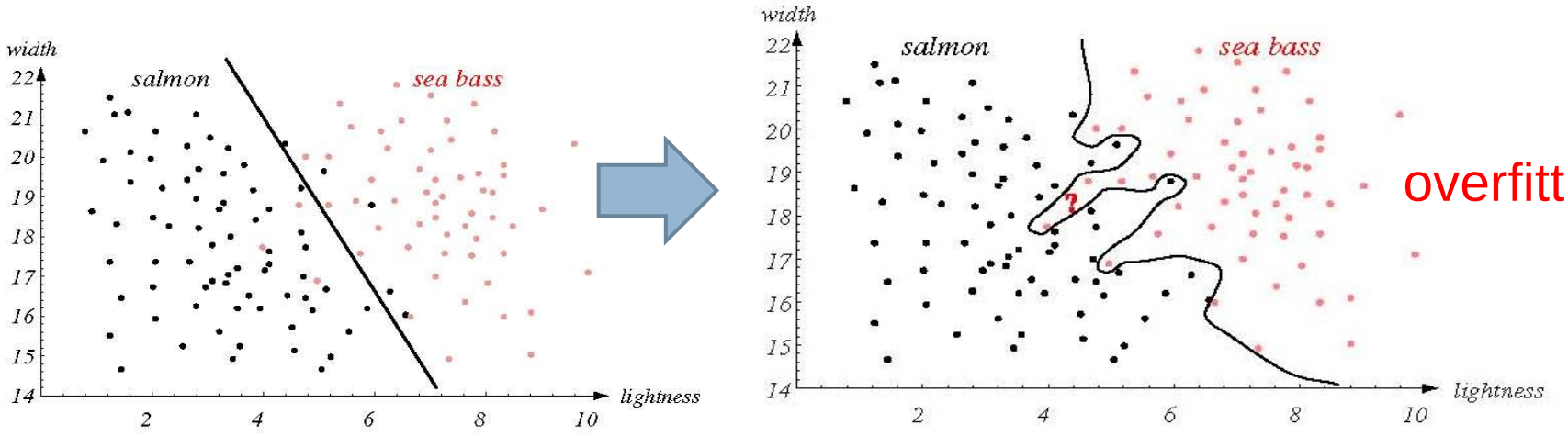
- If the classes are linearly not separable then partition the *feature space* into two regions by finding the decision boundary that minimizes the error.



- How should we find the optimal decision boundary?

Design of classifier- complex model

- We can get perfect classification performance on the training data by choosing complex models.
- Complex models are tuned to the particular training samples, rather than on the characteristics of the true model.



How well can the model generalize to unknown samples?

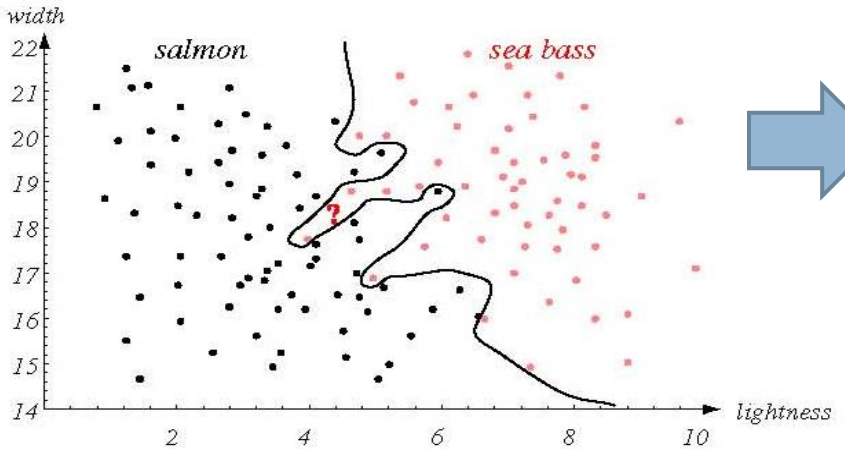
Design of classifier- complex model

- If an algorithm works well on the training set but fails to generalize on test data (unseen data), it is called the **problem of generalization or overfitting.**

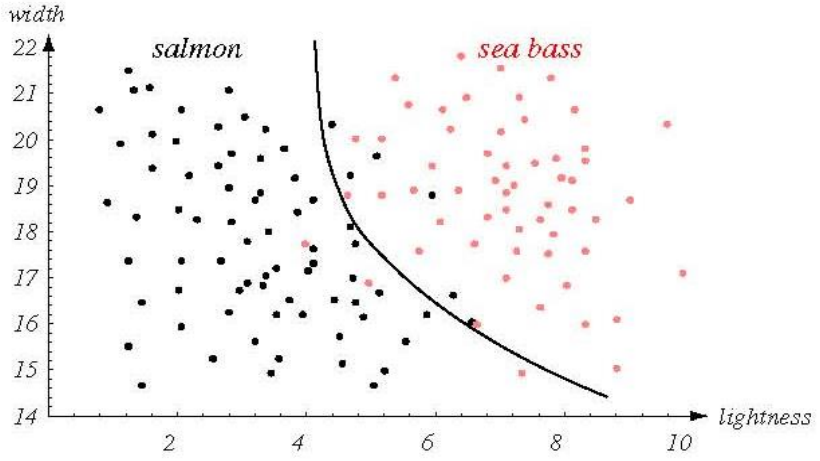
Generalization

- Generalization is defined as the ability of a classifier to produce correct results on unseen (novel) patterns.
- How can we improve generalization performance ?
  - More training examples (i.e., better model estimates).
  - Simpler models usually yield better performance.

complex model



simpler model



Generalization - example

- 12 total breeds of dogs within those pictures
- Out of 10 are in training and 2 for testing

Training Data

Testing Data

Generalization - example

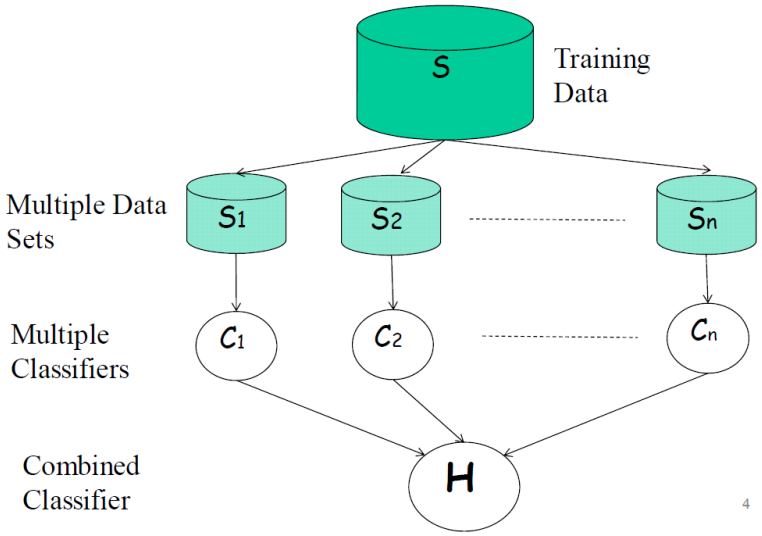
- This concept of learning from some data and **correctly** applying the gained knowledge on other data is called **generalization.**

Generalization

- Many techniques are available to improve generalization
  - Regularization- L1 and L2
  - Dropout
  - Early stopping

Design of classifier

- **Ensembles of Classifiers**
- Performance can be improved using a "pool" of classifiers.
- How should we build and combine different classifiers ?



Summary

- Classifiers
- Linear and non-linear classifiers
- Supervised learning
- Unsupervised learning
- Reinforcement learning
- generalization

THANK YOU