ECE-175A

Elements of Machine Intelligence - I

Ken Kreutz-Delgado

(Nuno Vasconcelos)

ECE Department, UCSD

Winter 2012

The course

- The course will cover basic, but important, aspects of **machine learning and pattern recognition**

- We will cover a lot of ground, at the end of the quarter you'll know how to implement a lot of things that may seem very complicated today

- Homework & (Possible) Computer Assignments will count for 30% of the overall grade. These assignments will be graded "A" for effort.

- Exams: 1 mid-term, date TBA- 30%
  1 final – 40% (covers everything)

Resources

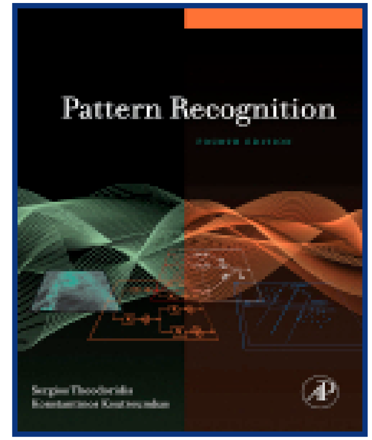
- Course web page is accessible from,
  http://dsp.ucsd.edu/~kreutz
  - All materials, except homework and exam solutions will be available there. Solutions will be available in my office “pod”.

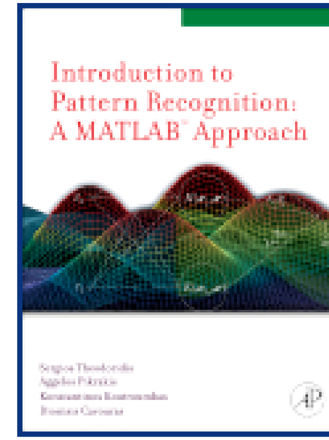
- Course Instructor:
  - Ken Kreutz-Delgado, kreutz@ece.ucsd.edu, EBU 1- 5605.
  - Office hours: Wednesday, Noon-1pm.

- Administrative Assistant:
  - Travis Spackman (tspackman@ece.ucsd.edu), EBU1 – 5600. Travis may sometimes be involved in administrative issues.

Required Textbooks:

- ***[Pattern Recognition](#) 4e***
  - S. Theodoridis & K. Koutroumbas
    Academic Press, 2009
- ***Introduction to Pattern Recognition: A Matlab Approach***
  - S. Theodoridis et al.
    Academic Press, 2010





Alternative reference texts:

- *Pattern Recognition and Machine Learning*, C.M. Bishop, Springer, 2007.
- *Pattern Classification*, Duda, Hart, Stork, Wiley, 2001

Prerequisites you ***must*** know well:

- **Linear algebra**, as in *Linear Algebra*, Strang, 1988
- **Probability and conditional probability**, as in *Fundamentals of Applied Probability*, Drake, McGraw-Hill, 1967

Why Machine Learning?

- **Good question!** After all there are many systems & processes in the world that <u>are</u> well-modeled by deterministic equations
  - E.g. $f = m\,a$; $V = I\,R$, Maxwell's equations, and other physical laws.
  - There are acceptable levels of "noise", "error", and other "variability".
  - In such domains, we don't need statistical learning.
- ***However***, **<u>learning is necessary</u>** ***when*** there is a need for ***predictions*** about, or ***classification*** of, ***poorly known* and/or *random* vector data Y**, that
  - represents important events, situations, or objects in the world;
  - which may (or may not) depend on other factors (variables) X;
  - is ***impossible or too difficult*** to derive an exact, deterministic model for;
  - might be an instantaneous snapshot of ***a constantly changing world***.

Examples and Perspectives

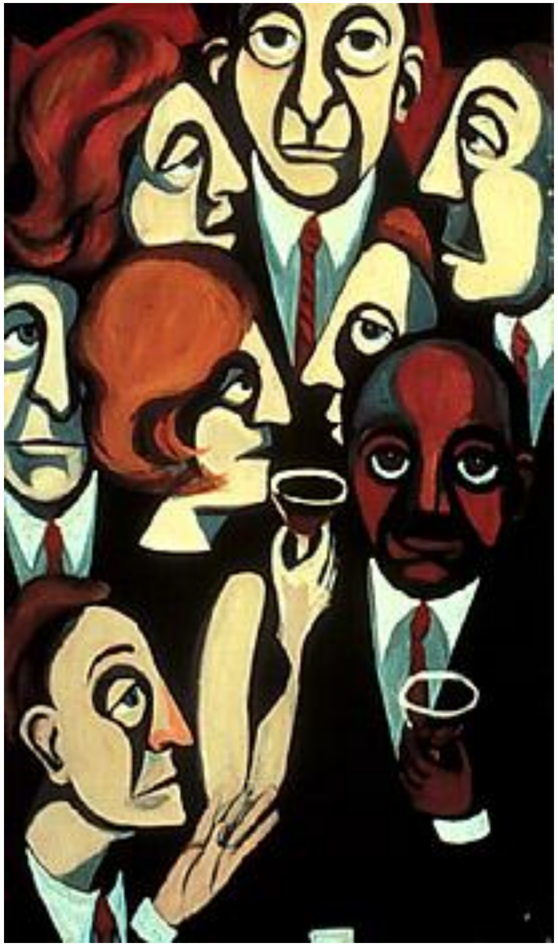
- ***The “Data-Mining” viewpoint:***
  - **HUGE** amounts of data that **does NOT follow deterministic rules**
  - E.g. given an history of thousands of customer records and some questions that I can ask you, how do I predict that you will pay on time?
  - Impossible to derive a theory for this, must be learned

While many associate learning with data-mining, it is by no means the only important application or viewpoint.

- ***The Signal Processing viewpoint:***
  - Signals combine in ways that depend on **“hidden structure”** (e.g. speech waveforms depend on language, grammar, etc.)
  - Signals are usually subject to significant amounts of **“noise”** (which sometimes means “things we do not know how to model”)

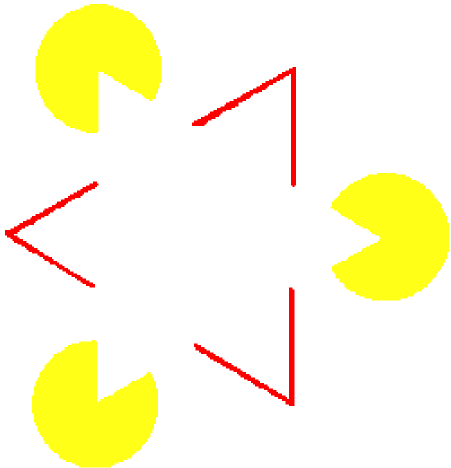
Examples - Continued

- Signal Processing viewpoint (Cont'd)
  - E.g. ***the Cocktail Party Problem*:**
    - Although there are all these people talking loudly at once, you can still understand what your friend is saying.
    - How could you build a chip to separate the speakers? (As well as your ear and brain can do.)
    - Model the hidden dependence as
      - a linear combination of independent sources + noise
  - Many other similar examples in the areas of wireless, communications, signal restoration, etc.



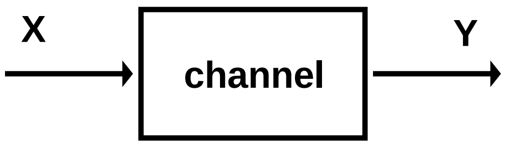
Examples (cont'd)

- ***The Perception/AI viewpoint:***
  - It is a complex world; one cannot model everything in detail
  - Rely on ***probabilistic models*** that explicitly account for the variability
  - Use the laws of probability to make inferences. E.g.,
    - P(burglar | alarm, no earthquake) is high
    - P(burglar | alarm, earthquake) is low
  - There is a whole field that studies "perception as ***Bayesian inference***"
  - In a sense, perception really is "confirming what you already know."
  - priors + observations = robust inference



Examples (cont'd)

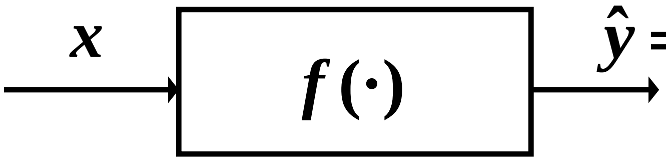
- ***The Communications Engineering viewpoint:***
  - **Detection** problems:



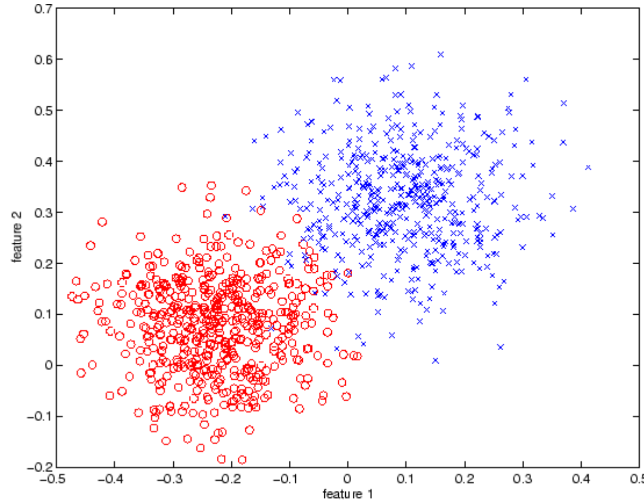
  - You observe Y and know something about the statistics of the channel. What was X?
  - This is the canonical *detection problem.*
  - For example, face detection in computer vision: "I see pixel array Y. Is it a face?"

What *is* Statistical Learning?

- **<u>Goal</u>:** Given a **relationship between** a **feature vector *x*** and a **vector *y***, and **iid data samples (x_i,y_i),** find an **approximating function $f(x) \approx y$**



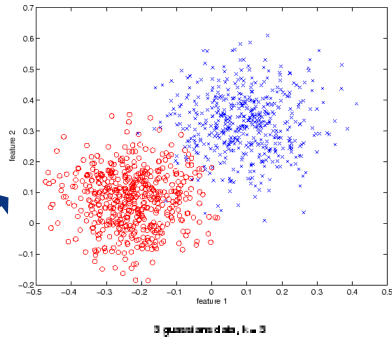
$$\hat{y} = f(x) \approx y$$

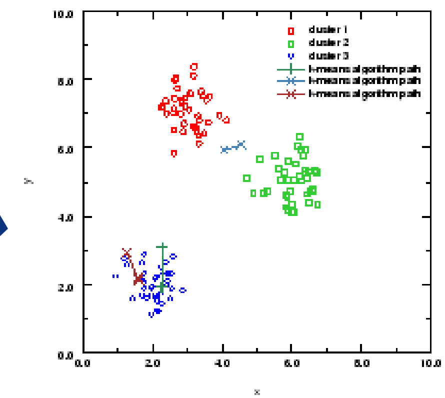


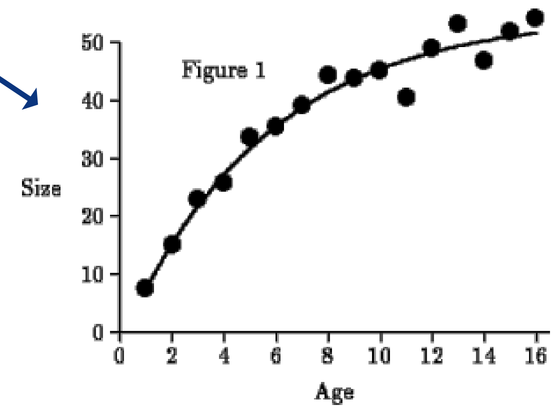
- This is called **training** or **<u>learning</u>**.
- **<u>Two major types</u>** of learning:
  - **<u>Unsupervised</u> Classification (aka <u>Clustering</u>)** or **Regression ("blind" curve fitting):** only ***X*** is known.
  - **<u>Supervised</u> Classification** or **Regression:** both ***X*** and target value ***Y*** are known during training, only ***X*** is known at test time.

Supervised Learning

- X can be anything, but the ***type*** of ***known data*** Y dictates the type of supervised learning problem
  - Y in {0,1} is referred to as ***Detection*** or ***Binary Classification***
  - Y in {0, ..., M-1} is referred to as ***(M-ary) Classification***
  - Y continuous is referred to as ***Regression***
- Theories are quite similar, and algorithms similar most of the time
- We will usually emphasize **classification**, but will talk about regression when particularly insightful

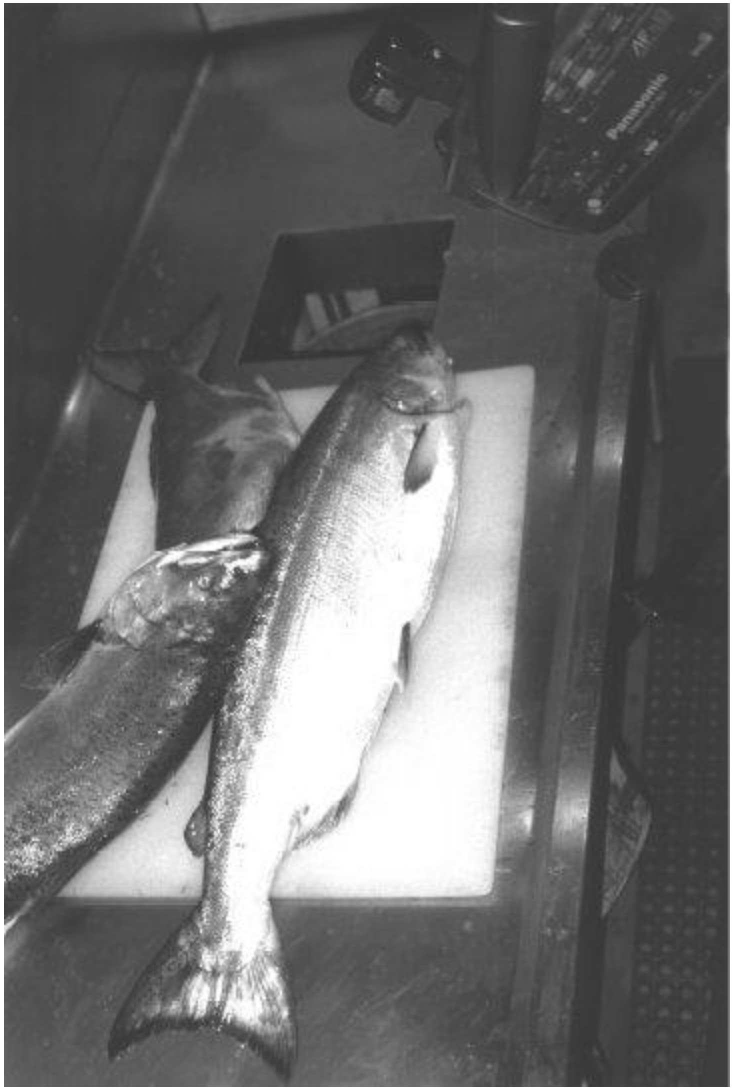






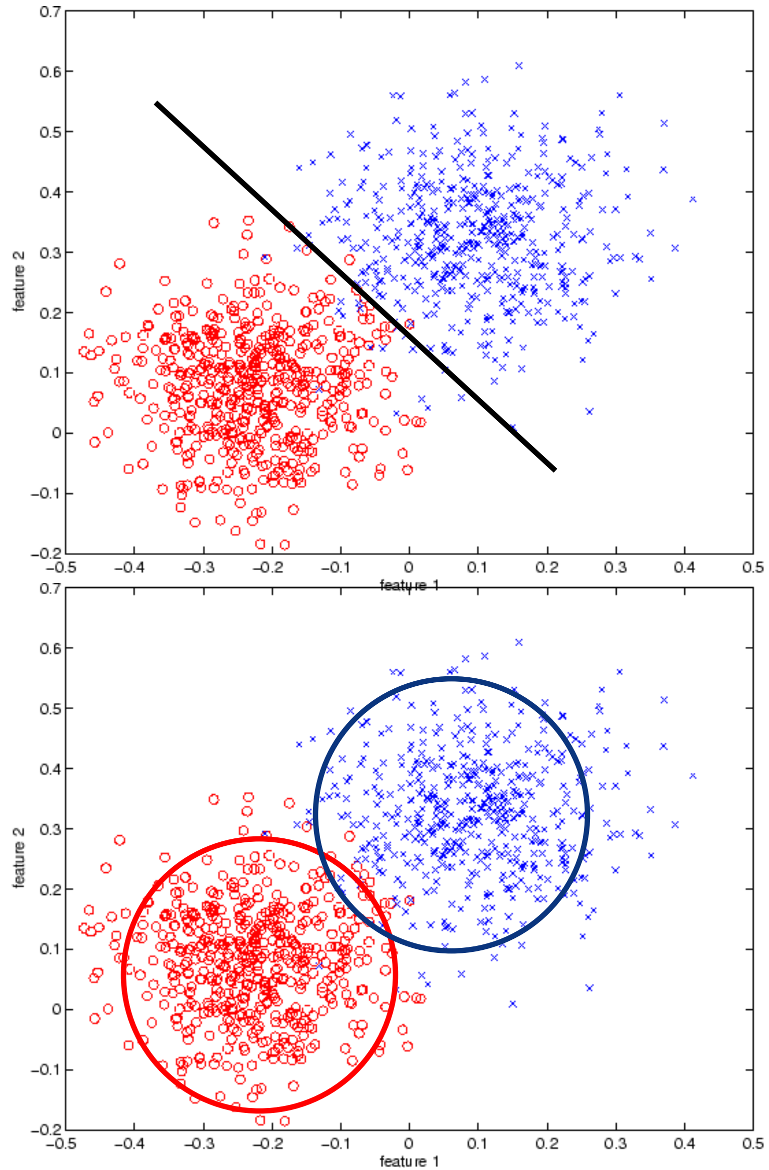
Example

- **Classification** of Fish:
  - Fish roll down a conveyer belt
  - Camera takes a picture
  - Decide if is this a salmon or a sea-bass?
- ***Q1***: **What *is* X?** E.g. what ***features*** do I use to distinguish between the two fish?
- This is somewhat of an art-form. Frequently, the best is to ask ***domain experts***.
- E.g., expert says use *overall length* and *width of scales*.



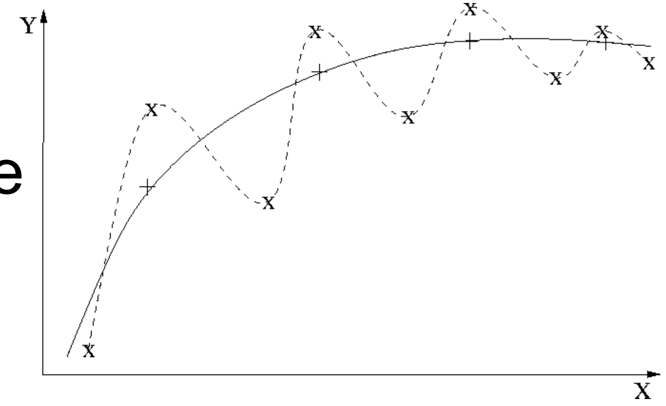
Q2: How to do Classification/Detection?

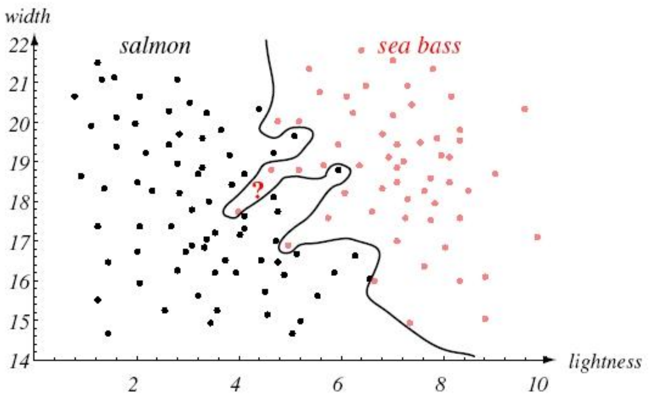
- **Two major types** of classifiers
- ***Discriminant***: determine the ***decision boundary in feature space*** that best separates the classes;
- ***Generative***: ***fit a probability model to each class*** and then compare the probabilities to find a decision rule.
- A lot more on the **intimate relationship between these two approaches** later!



Caution

- How do we know learning has worked?
- We care about ***generalization***, i.e. accuracy *outside* the training set
- Models that are "too powerful" on the training set can lead to ***over-fitting***:
  - E.g. in regression one can always *exactly* fit *n* pts with polynomial of order *n-1.*
  - Is this good? how likely is the error to be small outside the training set?
  - Similar problem for classification

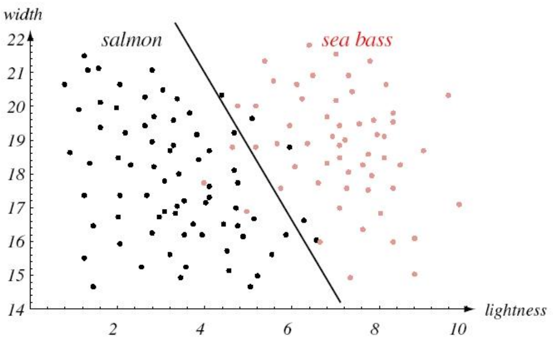


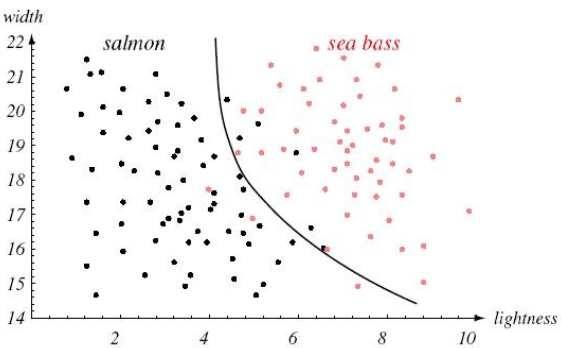


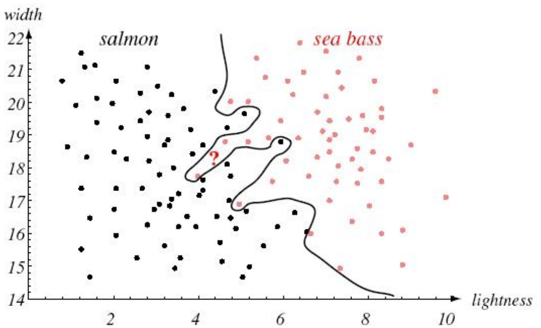
- Fundamental Rule: only ***hold-out test-set*** performance results matter!!!

Generalization

- Good generalization requires controlling the trade-off between training and test error
  - training error large, test error large
  - training error smaller, test error smaller
  - training error smallest, test error largest
- This trade-off is known by many names
- In the generative classification world it is usually due to the **bias-variance trade-off** of the class models

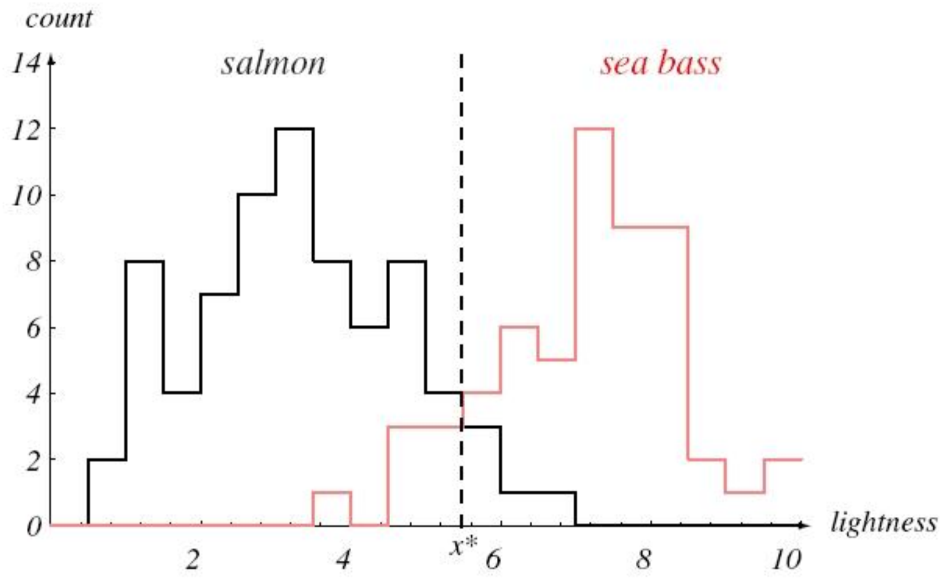






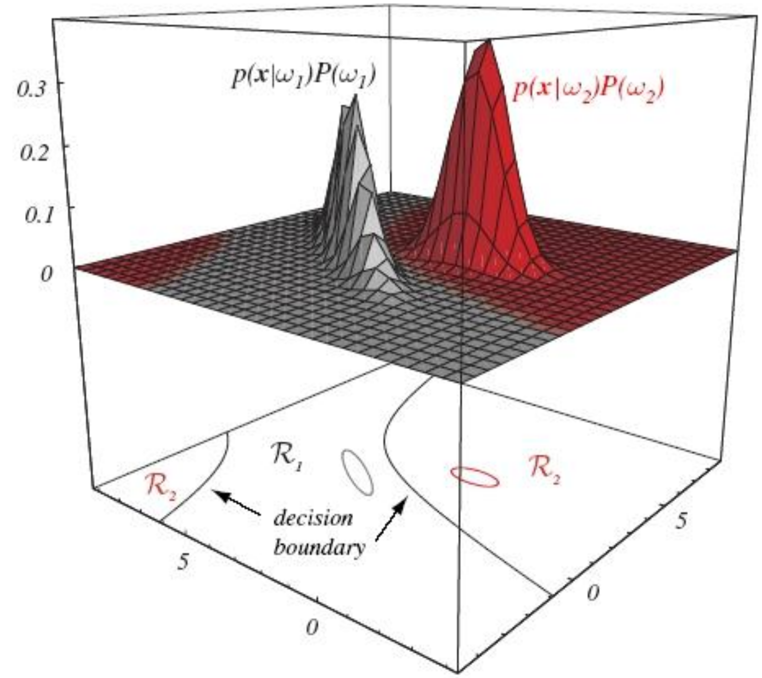
Generative Model Learning

- Each class is characterized by a probability density function (***class conditional density***), the so-called ***probabilistic generative model***. E.g., a Gaussian.
- **Training data is used to estimate the class pdfs.**
- Overall, the process is referred to as ***density estimation***
- A ***nonparametric approach*** would be to estimate the pdfs using ***histograms:***



Decision rules

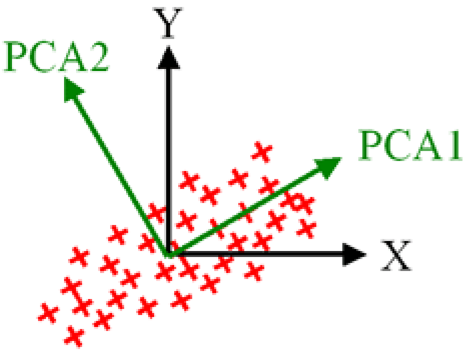
- **Given class pdfs, *<u>Bayesian Decision Theory</u>* (BDT) provides optimal rules for classification**

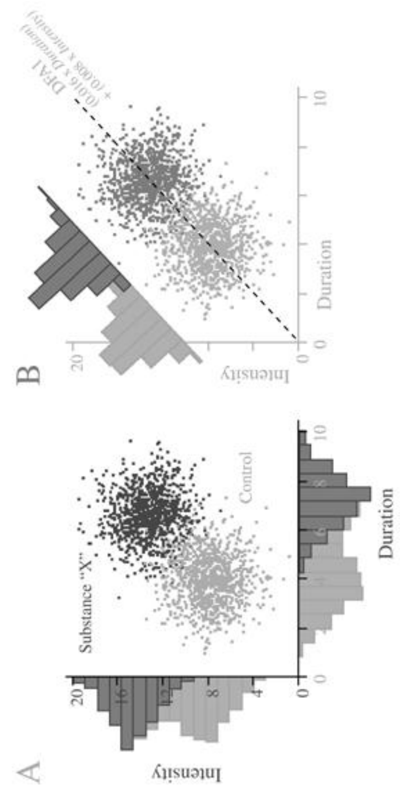


- "Optimal" here might mean **minimum probability of error**, for example
- We will
  - Study BDT in detail,
  - Establish connections to other decision principles (e.g. linear discriminants)
  - Show that Bayesian decisions are usually intuitive
- Derive optimal rules for a range of classifiers

Features and dimensionality

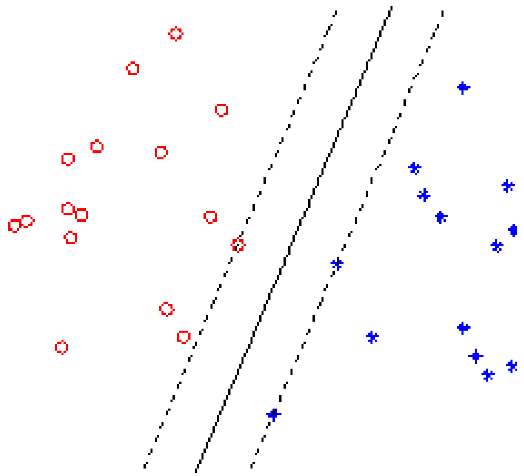
- For most of what we have said so far
  - Theory is well understood
  - Algorithms are available
  - Limitations can be characterized
- Usually, good features are an art-form
- We will survey traditional techniques
  - Bayesian Decision Theory (BDT)
  - Linear Discriminant Analysis (LDA)
  - Principal Component Analysis (PCA)
- and (perhaps) some more recent methods
  - Independent Components Analysis (ICA)
  - Support Vectors Machines (SVM)





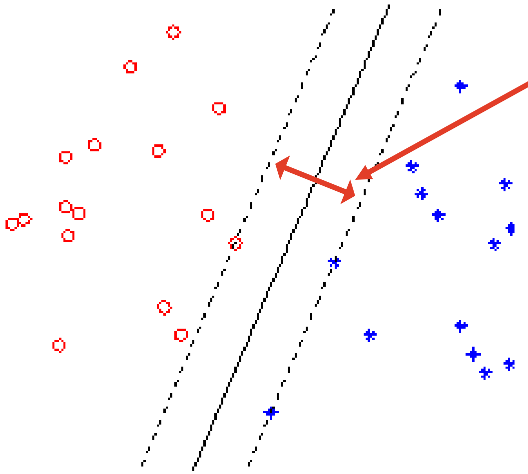
Discriminant Learning

- Instead of learning models (pdf’s) and deriving a decision boundary from the model, ***learn the boundary directly***
- There are many such methods. The simplest case is the so-called ***(separating) <u>hyperplane classifier</u>***
  - Simply find the hyperplane that best separates the classes, assuming ***linear separability*** of the features:



Support Vector Machines

- How do we do this efficiently in high-dimensional feature spaces?
- The most recently developed classifiers are based on the use of ***support vectors***.
  - One ***transforms the data into linearly separable features*** using ***kernel functions*.**
  - The best performance is obtained by maximizing the margin
  - This is the distance between decision hyperplane and closest point on each side

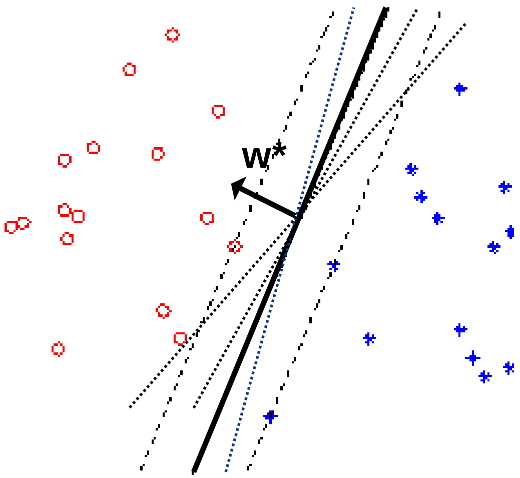


Support Vector Machine (SVM)

- For ***separable classes***, the ***training error*** can be made zero by classifying each point correctly
- This can be implemented by solving the optimization problem

$$w^* = \arg\max_{w} \ \text{margin}(w)$$

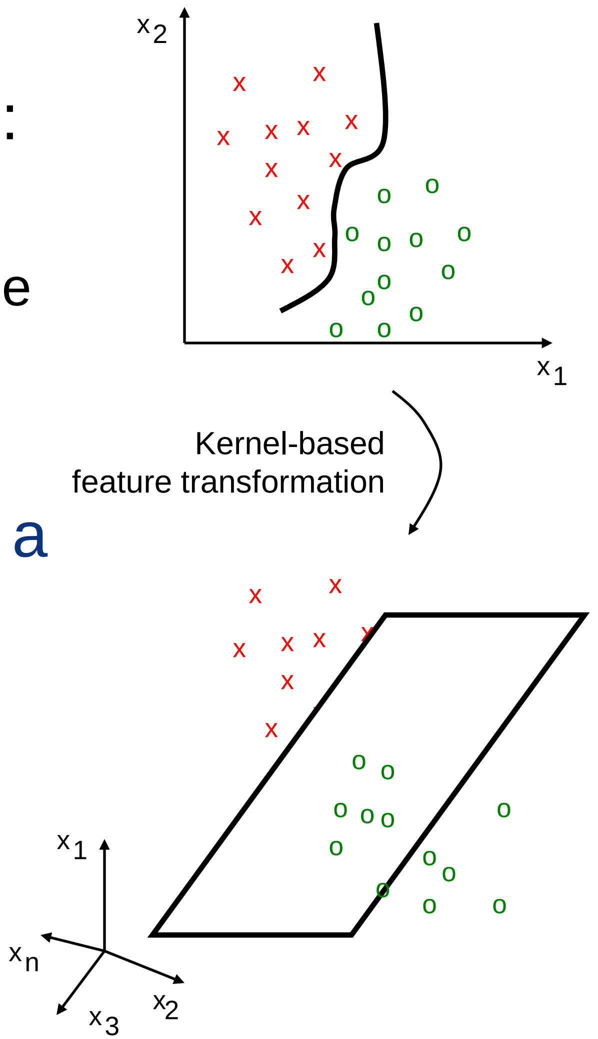
$$\text{s.t. } x_l \text{ is correctly classified } \forall l$$



- This is an optimization problem with many constraints, not trivial but solvable
- The resulting classifier is the "***<u>support-vector machine</u>***" The points on the margin are the "support vectors".

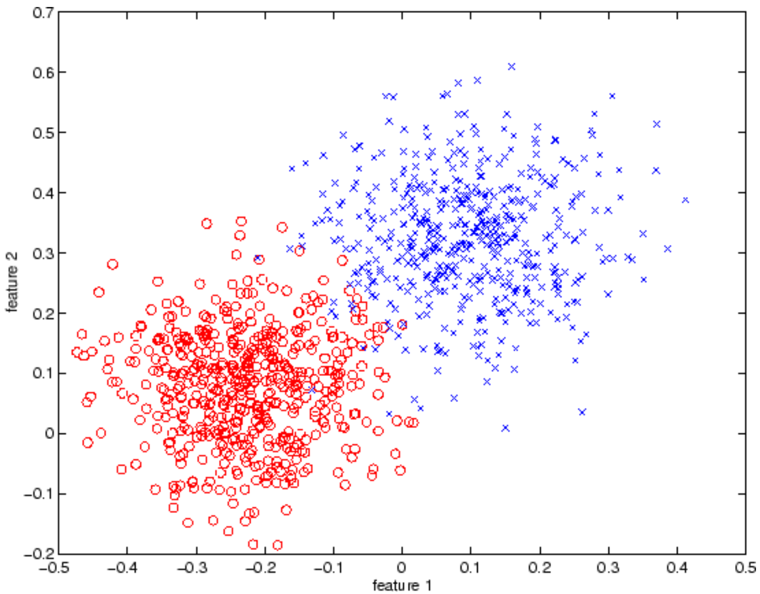
Kernels and Linear Separability

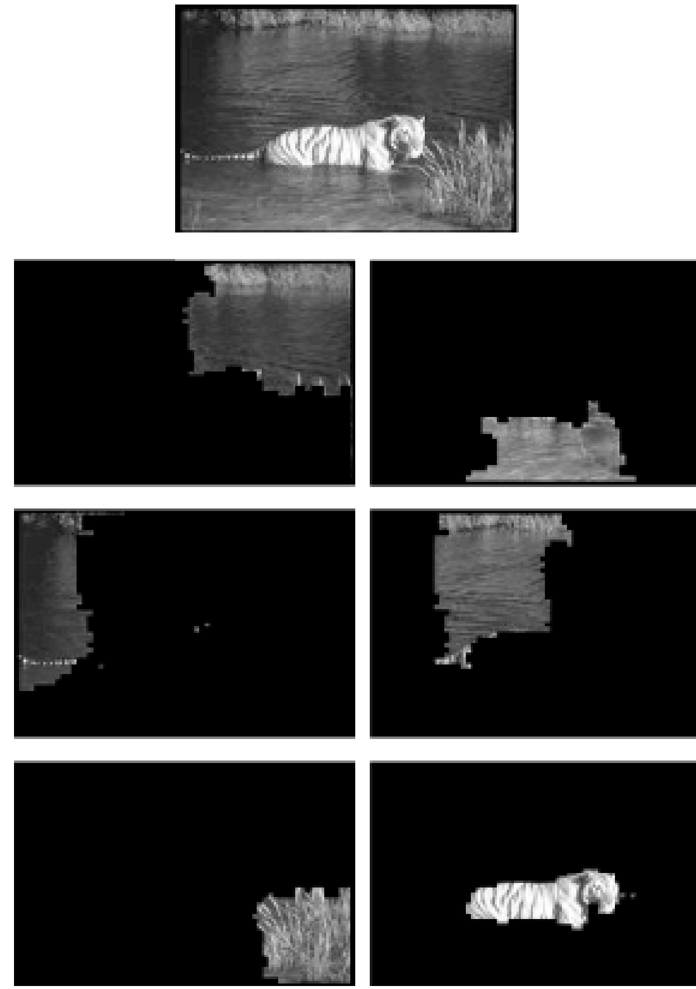
- The ***trick*** is to map the feature space to a higher dimensional feature space:
  - Non-linear boundary in original space
  - Becomes hyperplane in transformed space
- This can be done efficiently by the introduction of a ***kernel function***
- Classification problem is mapped into a reproducing kernel Hilbert space
- **Kernels are at the core of the success of SVM classification**
- Most classical linear techniques (e.g. PCA, LDA, ICA, etc.) can be kernel-ized with significant improvement



Unsupervised learning

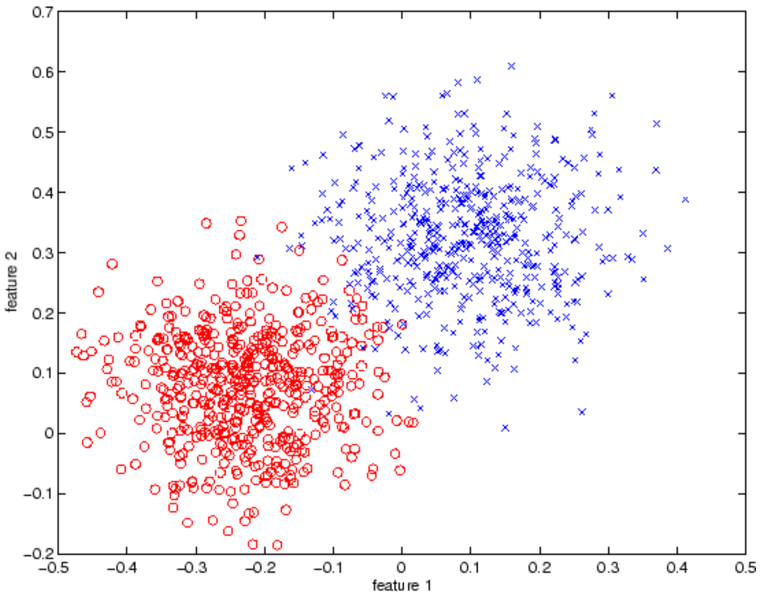
- So far, we have talked about **supervised learning:**
  - We know the class of each point
- In many problems this is not feasible to do (e.g. image segmentation)





Unsupervised learning

- In these problems we are given X, but ***not*** Y
- The ***standard algorithms for this are <u>iterative</u>***:
  - Start from best guess
  - Given Y-estimates fit class models
  - Given class models re-estimate Y-estimates



- This "boot-strap" procedure usually converges to a locally optimal solution, although not necessarily the global optimum
- Performance worse than that of supervised classifier, but this is the best we can do.

Reasons to take the course

- **To learn about Classification and Statistical Learning**
  - tremendous amount of theory
  - but things invariably go wrong
  - too little data, noise, too many dimensions, training sets that do not reflect all possible variability, etc.
- **To learn that good learning solutions require:**
  - knowledge of the domain (e.g. “these are the features to use”)
  - knowledge of the available techniques, their limitations, etc.
  - In the absence of either of these, you will fail!
- **To learn skills that are *highly valued* in the marketplace!**

END