ECE-271A
Statistical Learning I

Nuno Vasconcelos

ECE Department, UCSD

The course

- the course is an introductory level course in statistical learning
- by introductory I mean that you will not need any previous exposure to the field, not that it is basic
- we will cover the foundations of Bayesian or generative learning
- 271B is a follow-up course on discriminant learning
- more on generative vs discriminant later
- 271C is a follow-up course on deep learning

Logistics

- Exams: 1 mid-term - 35%
  1 final – 45% (covers everything)

- Quizzes (20%):
  - one problem every week.
  - a small computational problem. By small, I mean in terms of concepts, thinking, etc.
  - some computational problems will require a fair amount of computer power, e.g. a few hours on a laptop.
  - be sure to start early
  - will count 20%, but almost impossible without it.
  - will give you the hands-on experience needed to be able to claim that you really know learning!

Homework policies

- homework is **individual**
- two types of problems
- ungraded
  - you will be given the solutions
  - OK to discuss with anyone
- graded – take home **Quizzes**
  - think of these as part of an exam
  - don't ask questions you would not ask on exam room
  - don't ask "does my result look OK?," or "did anyone get something like this?"
- quizzes are due on the dates specified on the website.

Academic integrity

- is taken very seriously at UCSD and in this course
- all work done for grade must be **individual**
- not allowed to
  - talk to friends or classmates about graded problems
  - use **any** ECE271A material not explicitly handed out by us
  - this includes consulting **any** websites other than the class web site, piazza, or canvas
- be aware that
  - we refer violations to UCSD Academic Integrity Office
  - penalty ranges from F in the class to expulsion from UCSD
  - despite these warnings there are always 3-4 cases a year

Quizzes

- statistical learning only makes sense when you try it on data
- we will test what we learn on an image processing problem
  - given the cheetah image, can we teach a computer to segment it into object and foreground?
  - the question will be answered with different techniques, typically one problem per week
  - a total of 4 computer problems
- keep an eye on the big picture, e.g. "did this improve over what we had done before?"

Resources

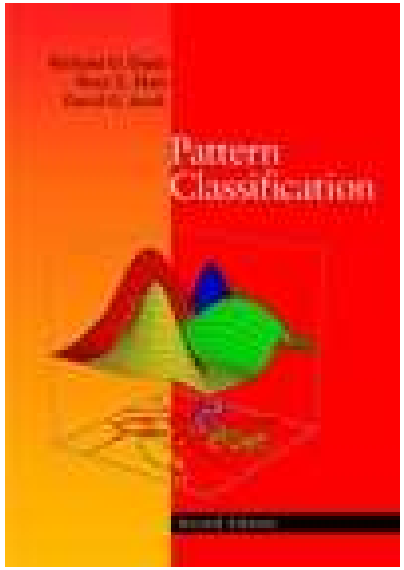
- Course web page: http://ucsd.svcl.ucsd.edu/~nuno
  - Accessible from Canvas page
  - all handouts, problem sets, code available there

- TA: see canvas

- Me: Nuno Vasconcelos, nuno@ece.ucsd.edu, EBU1-5603

- Office hours:
  - TA: see canvas
  - mine: TBA
  - for homework talk to TAs, my OHs for other issues

Texts



- we follow:
  - "Pattern Classification", Duda, Hart, and Stork, John Willey and Sons, 2001
  - will follow closely, hand-outs where needed
- various other good, but optional, texts:
  - "Pattern Recognition and Machine Learning", Bishop, 2006
  - "Elements of Statistical Learning", Hastie, Tibshirani, Fredman, 2001
  - "Bayesian Data Analysis", Gelman, Rubin, Stern, 2003.
  - "A Probabilistic Theory of Pattern Recognition", Devroye, Gyorfi, Lugosi, 1998 (more than what we need)
- stuff you must know well:
  - "Linear Algebra", Gilbert Strang, 1988
  - "Fundamentals of Applied Probability", Drake, McGraw-Hill, 1967

The course

- why statistical learning?
- there are many processes in the world that are ruled by deterministic equations
  - e.g. f = ma; linear systems and convolution, Fourier, etc, various chemical laws
  - there may be some "noise", "error", "variability", but we can leave with those
  - we don't need statistical learning
- learning is needed when
  - there is a need for predictions about world variables Y
  - that depend on factors (other variables) X
  - in a way that is impossible or too difficult to derive an equation for.

Examples

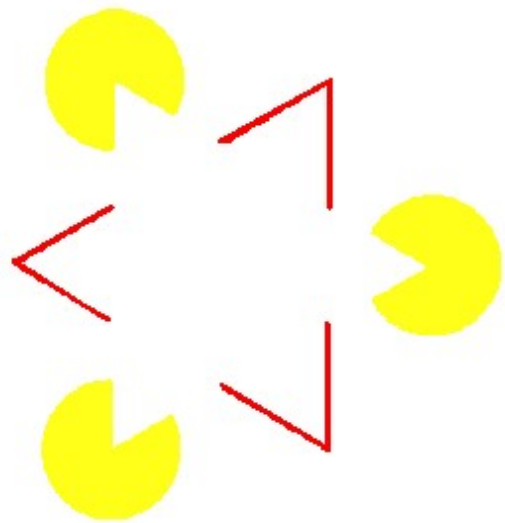
- data-mining view:
  - data that does not follow deterministic rules
  - e.g. given an history of thousands of customer records and some questions that I can ask you, how do I predict that you will pay on time?
  - impossible to derive a theorem for this, must be learned
- while many associate learning with data-mining, it is by no means the only or more important application
- signal processing view:
  - signals combine in ways that depend on "hidden structure" (e.g. speech waveforms depend on language, grammar)
  - signals are usually subject to significant amounts of "noise" (sometimes means "things we do not know how to model")

Examples (cont'd)

- signal processing view:
  - e.g. the cocktail party problem,
  - although there are all these people talking, I can figure everything out.
  - how do I build a chip to separate the speakers?
  - model the hidden dependence as
    - a linear combination of independent sources
    - noise
  - many other examples in the areas of wireless, communications, signal restoration, etc.

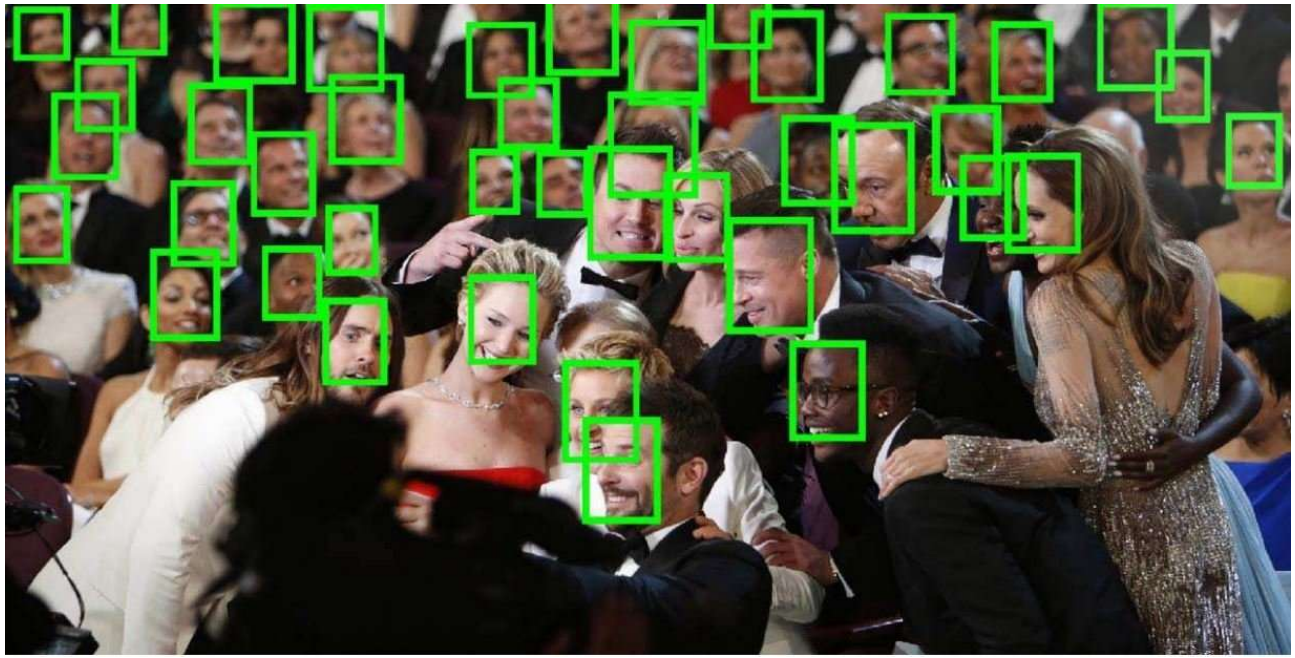
Examples (cont'd)

- perception/AI view:
  - it is a complex world, I cannot model everything in detail
  - rely on probabilistic models that explicitly account for the variability
  - use the laws of probability to make inferences
  - a whole field that studies "perception as Bayesian inference"
  - perception just confirms what you already know
  - priors + observations = robust inference



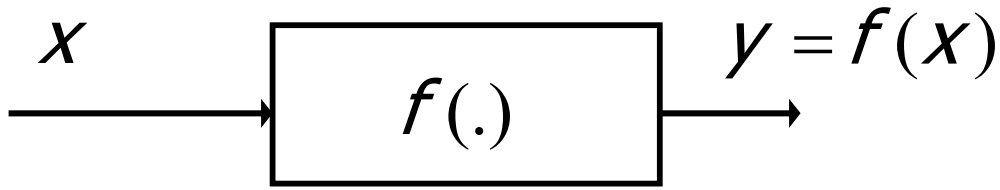
Examples (cont'd)

- communications view:
  - detection problems : I see Y and know something about the statistics of the channel. What was X?
  - canonic detection problem that appears all over learning.
  - for example, face detection in computer vision: "I see pixel array Y. Is it a face?"

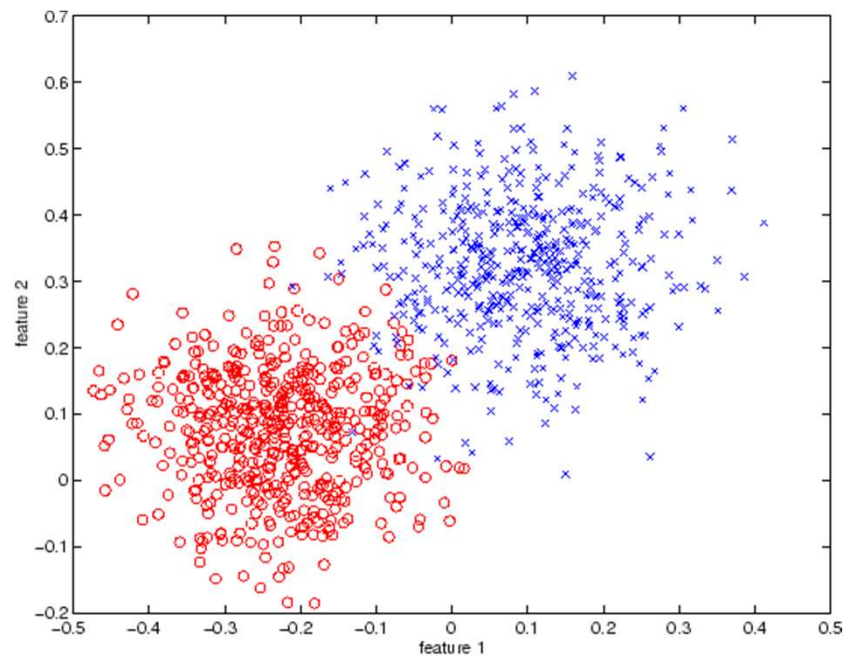


Statistical learning

- goal: given a function



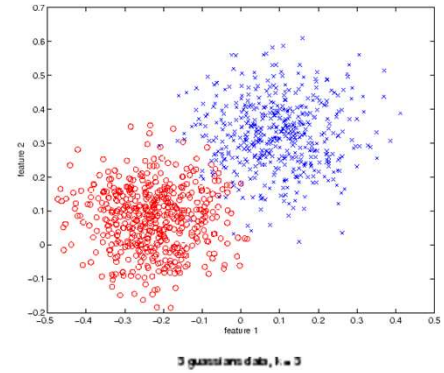
  and a collection of example data-points, learn what the function *f(.)* is.

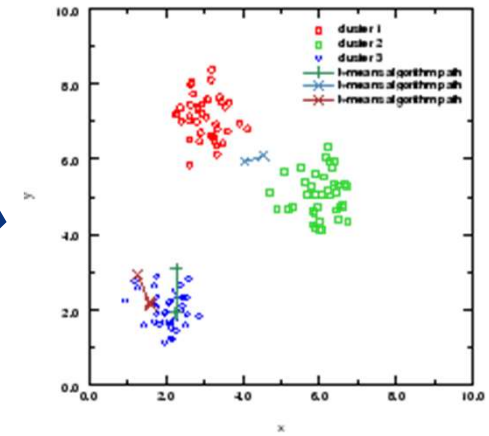


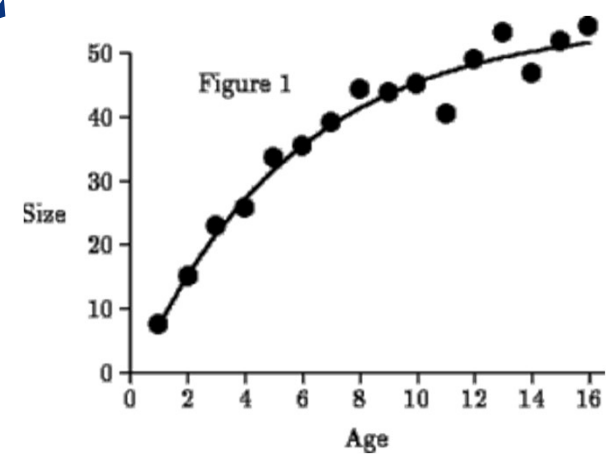
- this is called training.
- two major types of learning:
  - unsupervised: only X is known, usually referred to as clustering;
  - supervised: both are known during training, only X known at test time, usually referred to as classification or regression.

Supervised learning

- X can be anything, but the type of Y dictates the type of supervised learning problem
  - Y in {0,1} referred to as detection
  - Y in {0, ..., M-1} referred to as classification
  - Y real referred to as regression
- theory is quite similar, algorithms similar most of the time
- we will emphasize classification, but will talk about regression when particularly insightful





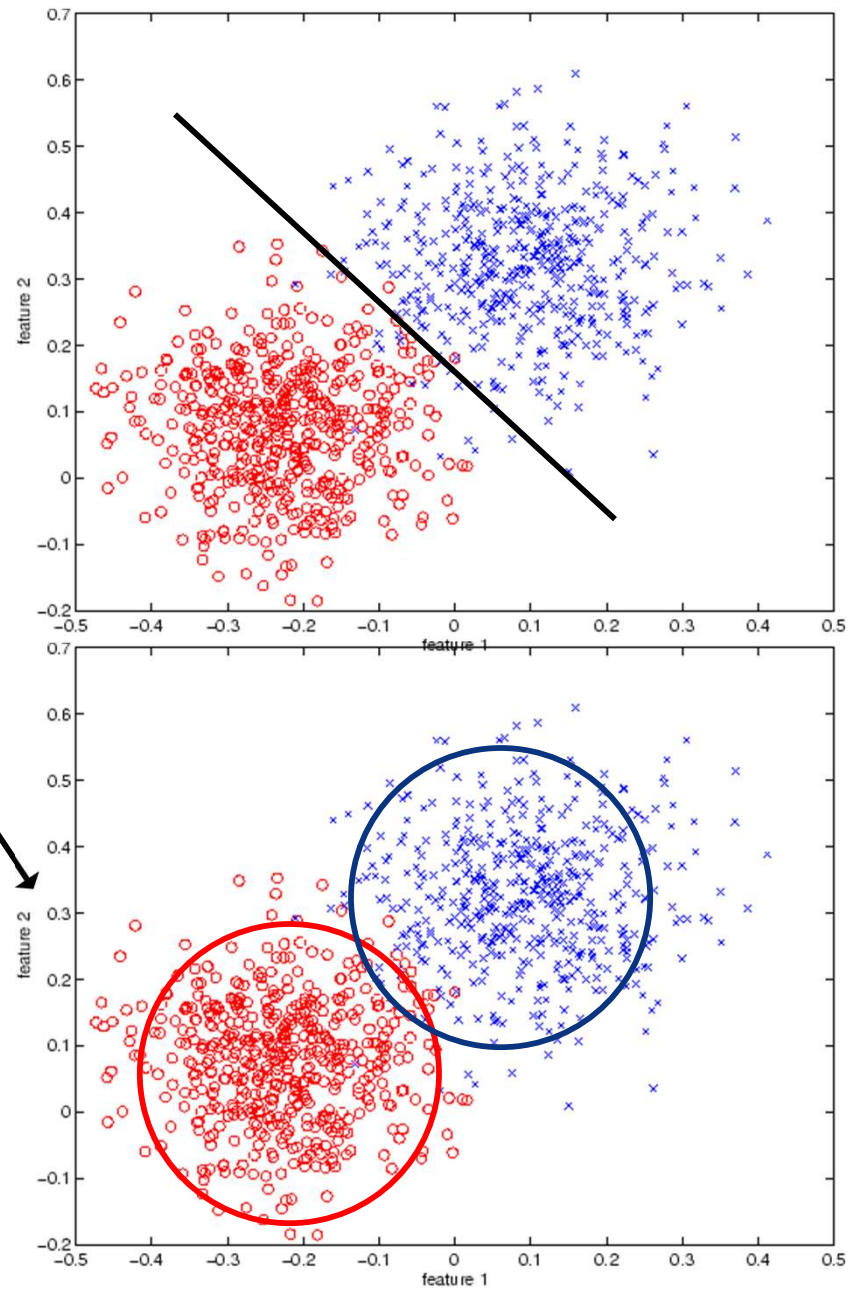


Example

- classifying fish:
  - fish roll down a conveyer belt
  - camera takes a picture
  - goal: is this a salmon or a sea-bass?
- Q: what is X? What features do I use to distinguish between the two fish?
- this is somewhat of an art-form. Frequently, the best is to ask experts.
- e.g. "obvious! use length and scale width!"

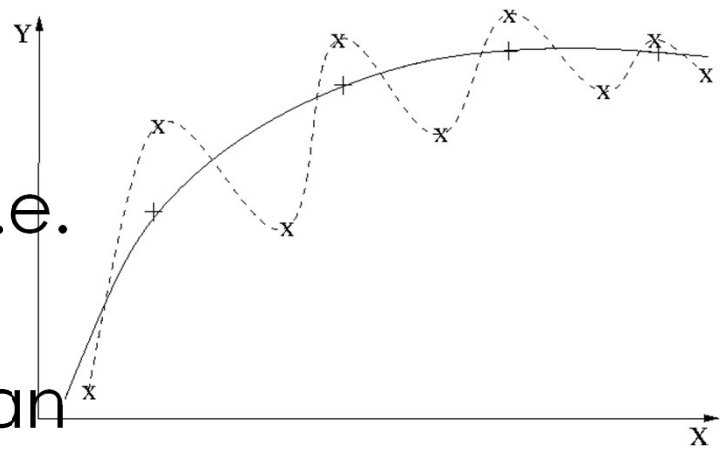
Classification/detection

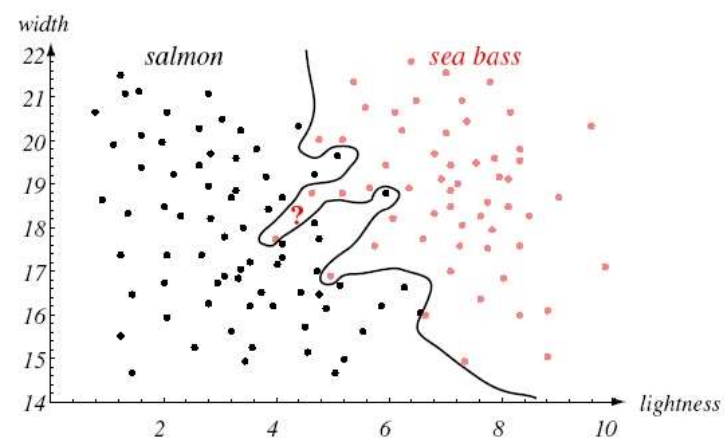
- two major types of classifiers:
  - discriminant: directly recover the decision boundary that best separates the classes;
  - generative: fit a probability model to each class and then "analyze" the models to find the border.
- a lot more on this later!
- focus will be on generative learning.
- discriminant will be covered by 271B.



Caution

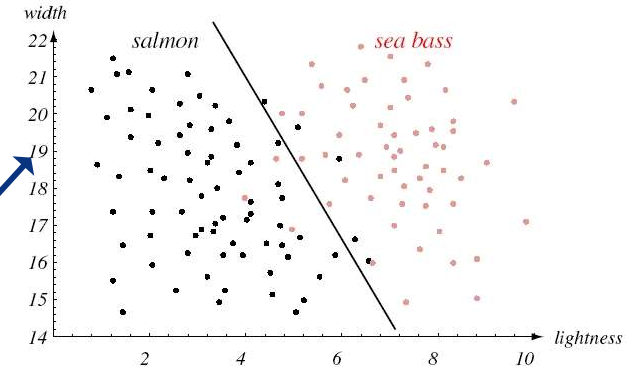
- how do we know learning worked?
- we care about generalization, i.e. accuracy outside training set
- models that are too powerful can lead to over-fitting:
  - e.g. in regression I can always fit exactly *n* pts with polynomial of order *n-1*.
  - is this good? how likely is the error to be small outside the training set?
  - similar problem for classification
- fundamental LAW: only test set results matter!!!

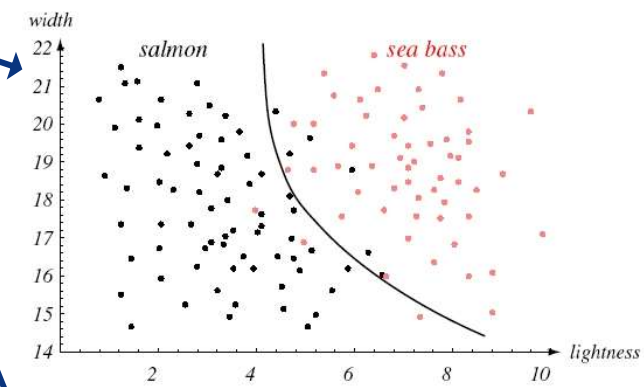


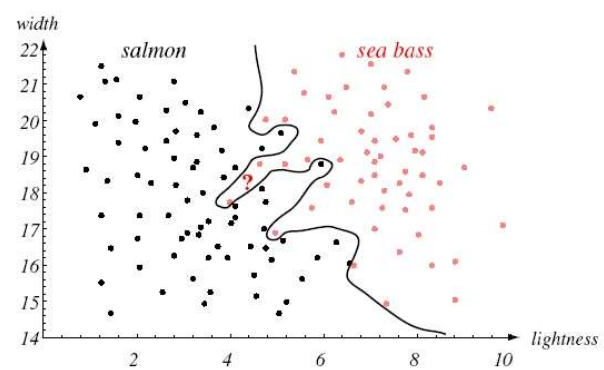


Generalization

- 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
- will look at this in detail







Any questions?