Lecture 2 Spring 2016

HW 1 due Thursday, April 7
1:30 PM

Submit
- Electronically
  - → CCLE
  - → Email reader*
    (* If you are not yet enrolled)
- In class Wed
- My office, Student Lounge
  8105/8105E

OH: Allie 8105 MSci/Student Lounge
11-12:50

Joey 9401 Boelter
- Monday 10-12
- Wednesday 11-12

Reading: Textbook
- 2.4 3.1-3.4

Last time!

I Machine Learning:

- What?
- Why?
- How?
- Why now?
- Why difficult?

II Supervised Learning

- Associations
- Classification
- Regression

III Unsupervised Learning

Today

I Classification

- More details

II. Memory

- Prediction, Generalization

III VC Dimension

IV Probabilistic view

- Maximum Likelihood

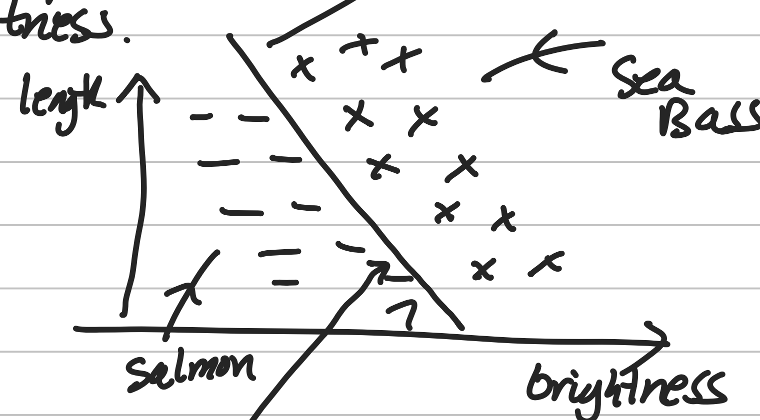
Classification. Linear Classifier

Task: classify fish as salmon or sea bass

What features to use?
choices : length of fish
width of fish
brightness (dark/bright)
texture
shape of head.

linear classifier

Use length and brightness.



Training Data

$\{(x_1^\mu, x_2^\mu, y^\mu) : \mu = 1 \text{ to } N\}$

$y = +1$, sea bass
$y = -1$, salman
x_1 - brightness, x_2 - length.

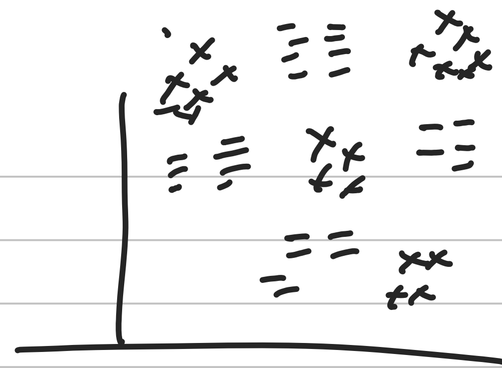
Want simple rule to discriminate between salmon and sea bass.

linear classifier sea bass on one side
salmon on the other.

Note: linear classifier / perceptron is another of the three classic machine learning methods.

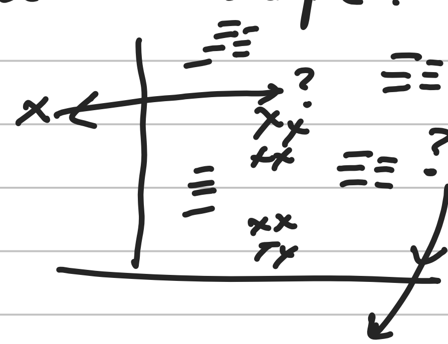
The third classic machine learning method is the nearest neighbor classifier.

Nearest neighbor classifier



Suppose we have training data.
we cannot find a linear classifier
that separates the ++ and --
examples.

Nearest neighbor classifies a new example?
by the nearest examples.



Note: the three classic methods
tend to be good for different types
of data. But nearest neighbor is
very good in general.

We will discuss these methods, and many others,
in the rest of the course.

Now we introduce a key concept in
Machine Learning. The difference between

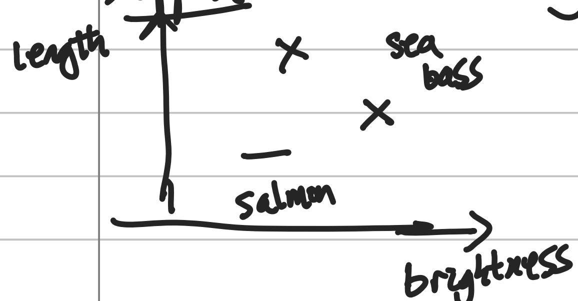
Memorization: finding a classifier that
gives good results on the training data.

Generalization: finding a classifier that
gives good results on data that
we haven't seen. Good prediction.

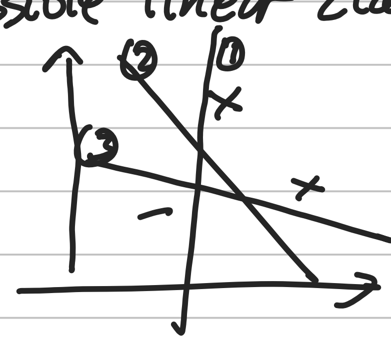
Memorizaton and Generalization.

We want to learn a classifier that works on data we have not seen yet.

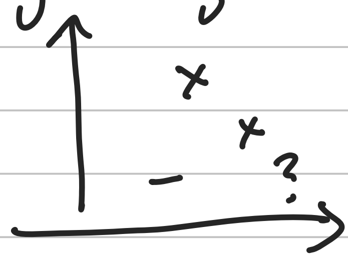
Suppose our training set contains only three examples



Many possible linear classifiers 1,2,3



But these three classifiers will not generalize to new data.



How to classify new data?

1. says ? is sea bass
2. says ? is sea bass
3. says ? is salmon.

Which is right?

Answer: we do not know. We do not have enough training data to learn the classifier. We need more data.

Memorization: All three classifiers (1), (2), (3) can classify the training data (i.e. memorize it)

Key Factors: amount of training data.
complexity of classifiers.

Need to ensure that complexity of classifiers $\ll$ amount of data.

Other Issues

Much of Machine Learning involves learning classifiers, or probabilities.

But we may also want to perform:

- Knowledge Extraction — use algorithms to understand the structure of the data.
- Compression — learn simple ways to describe the data.
- Outlier Detection — find instances which do not obey the rules, which are outliers
  These may signal the onset of fundamental changes — financial crises, wars, ..
  They may also signal events like fraud.

Another key issue is the curse of dimensionality.
Most machine learning problem involve high-dimensional data. E.g. $(x_1 \ldots x_{100}, y)$ not (x_1, x_2, y) (as in our examples)

Problem: • our geometric intuitions are bad in high-dimension.
• classifiers in high-dimensions may require an enormous amount of data.

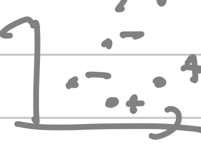
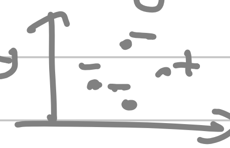
What happens if we have an infinite set of rules? – e.g. the set all seperating planes $ax+by+c=0$

The Vapnik–Chervonenkis VC dimension give a finite measure of the capacity of a hypothesis class A.

Introduce the concept of shattering.

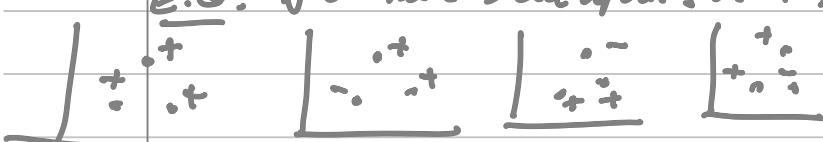
Suppose we have n data examples (features / attributes) $\{\underline{x}_i : i = 1..n\}$ in d-dim space. With general position assumption (data doesn't lie on a lower-dimensional subspace).

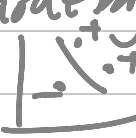
They are 2^n possible dichotomies of the data – separating the examples into two classes, positive and negative

 ← one dichotomy → another  and another → 

A set A of classifiers, shatters n examples in d-dim space if, for all dichotomies of the data, we can find a classifier in A which classifies the data correctly.

E.G. If we have 3 datapoints in 2D, there are $2^3 = 8$ dichotomies.

 and some examples with + replaced by –

For each dichotomy, we can find a separating plane which classifies the data perfectly → eg 

Hence, we know that we can classify the data perfectly before we even look at it.

The VC-dimension of a hypothesis class A is the maximum number of points that can be shattered. Note: this depends on the dimension of the space

For seperating hyperplanes, the VC dimension $= d+1$ (d = dim of space). i.e. VC = 3 for planes in 2D space.

This concept enables us to prove theorems for hypothesis spaces with finite VC dimension, but infinite number of classifiers (e.g. planes)

PAC Learning
Advacnced
for MLearn
2016

For example,

with prob $> 1-\delta$

$$R(\alpha) \leq R_{emp}(\alpha:N) + \sqrt{\frac{h(\log 2N/h) - \log \delta/4}{N}} \quad \text{for all } \alpha \in A$$

PAC Theorem — where h is the VC dimension of A, N is the total amount of data.

<u>Moral:</u> In order to generalize, you have to restrict the complexity (i.e. the VC dimension) of the set of classifiers you use by taking into account the amount of data

Bayes Decision Theory

How to make decisions in the presence of uncertainty?

<u>History</u>: 2nd World War
Radar for detection aircraft.
Codebreaking. Decryption.

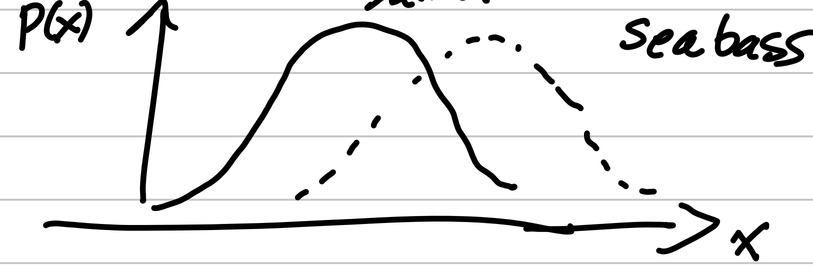
Observed Data $x \in \mathcal{X}$
State $y \in \mathcal{Y}$. <u>likelihood function</u>

$p(x|y)$ – conditional distribution
model how data is generated.

<u>Example</u> $y \in \{-1, 1\}$ Salmon / Sea Bass
Airplane / Bird

$$p(x|y) = \frac{1}{\sqrt{2\pi}\,\sigma_y} e^{-\frac{1}{2}\frac{(x-\mu_y)^2}{2\sigma_y^2}}$$ mean μ_y. variance σ_y^2.

<u>E.G.</u> x is length of fish.



(2) How to decide Sea Bass or Salmon?
Airplane or Bird

Maximum Likelihood (ML)

$$\hat{y}_{ML} = \text{ARG} \max_y p(x|y) \quad \left(p(x|\hat{y}_{ML}) \geq p(x|y) \ \forall y \right)$$

If $p(x|y=1) > p(x|y=-1)$ decide $y=1$
otherwise $y=-1$

Equivalently $\log \dfrac{p(x|y=1)}{p(x|y=-1)} > 0$ log-likelihood test.

Seems reasonable, but what if birds are more likely than airplanes?

Must take into account the prior probability $p(y=1), p(y=-1)$.

Bayes Rule $p(y|x) = \dfrac{p(x|y)p(y)}{p(x)}$

prob of y conditioned on observation.

If $p(y=1|x) > p(y=-1|x)$ decide $y=1$
otherwise decide $y=-1$

Maximum a Posteriori (MAP) $\hat{y}_{MAP} = \text{ARG} \max_y p(y|x)$

(3) **Another Ingredient**

→ what does it cost if you make a mistake?

i.e. suppose you decide $y = 1$, but really $y = -1$.

i.e. you may pay a big penalty if you decide it is a bird when it is a plane.

(Pascal's Wager: Bet on God)

Putting everything together.

likelihood function $p(x|y)$ $\quad x \in X, y \in Y$

prior $p(y)$

decision rule $\alpha(x)$ $\quad \alpha(x) \in Y$

loss function $L(\alpha(x), y)$ $\quad$ cost of making decision $\alpha(x)$ if true state is Y.

e.g. $L(\alpha(x), y) = 0$, if $\alpha(x) = y$

$L(\alpha(x), y) = 1$, if $\alpha(x) \neq y$, all errors penalized the same.

or $L(\alpha(x), y) = 0$, if $\alpha(x) = y$, $\quad$ PASCAL'S CASE.

$L(\alpha(x) = 1, y = -1) = 10$

$L(\alpha(x) = -1, y = 1) = 10{,}000{,}000{,}000{,}000$

$y = 1$, God exists, $y = -1$, God does not exist.

Examples of MAP Estimators--Lec 2 and 3
Helpful for homework

~~log~~

MAP Estimate. Gaussian case

$$\hat{y} = \arg\max \; p(x|y)\,p(y)$$

$\rightarrow$ Select $\hat{y} = 1$

$$\Leftrightarrow p(x|y=1)\underbrace{P(y=1)}_{P_1} \geq p(x|y=0)\underbrace{P(y=0)}_{P_0}$$

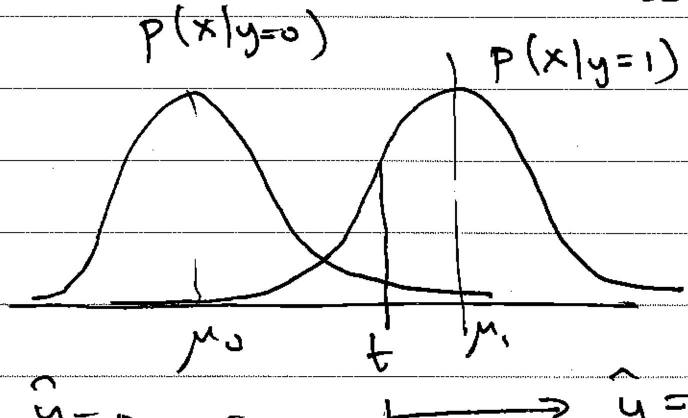
$$\Leftrightarrow \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(\cancel{y}-\mu_1)^2}{2\sigma^2}\right)P_1 \geq \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(\cancel{y}-\mu_0)^2}{2\sigma^2}\right)P_0$$

$$\Leftrightarrow \frac{(x-\mu_1)^2}{2\sigma^2} - \ln(p_1) \leq \frac{(\cancel{y}-\mu_0)^2}{2\sigma^2} - \ln(p_0)$$

$$\Leftrightarrow (x-\mu_1)^2 - (x-\mu_0)^2 \leq 2\sigma^2 \ln\left(\frac{P_1}{p_0}\right)$$

$$2(\mu_0-\mu_1)x + \mu_1^2 - \mu_0^2 \leq \quad ''$$

$$x \geq \underbrace{\frac{1}{2(\mu_1-\mu_0)}\left[2\sigma^2\ln\left(\frac{P_1}{p_0}\right) + \mu_1^2 - \mu_0^2\right]}_{t \,=\, \text{threshold.}}$$



Binary case

$$\hat{\alpha} = \arg\min_i \sum_j \underbrace{L(\alpha(x)=i, y=j)}_{C_{ij}} \underbrace{P(x, y=j)}_{P(x|y=j)P(y=j)}$$

Select $\hat{\alpha}=1$

$$\Leftrightarrow \; C_{10} P(x|y=0) P(y=0) + C_{11} P(x|y=1) P(y=1)$$
$$\leq C_{00} P(x|y=0) P(y=0) + C_{01} P(x|y=1) P(y=1)$$

Special case $C_{11} = C_{00} = 0$

$$\hat{\alpha}=1 \Leftrightarrow \underbrace{\frac{P(x|y=1)}{P(x|y=0)}}_{\text{Likelihood ratio}} \geq \frac{C_{10} P(y=0)}{C_{01} P(y=1)}$$

$\rightarrow$ Any binary Bayes' risk classifier is a likelihood ratio test (LRT).

Credit to Alan Yuille for some of notes