Lecture 11am Nov 4 2014

**Machine Learning -Supervised**

**-Unsupervised**

**Supervised** -- Use training data to infer a function. P(y | x)

-- Back Propagation

-- Perceptron

-- S.V.M.

**Unsupervised**  -- Use probabilistic way to solve (probability theory, Baye's Rule) P(x)

-- Clustering --K-means

-- Gaussian Mixture Model (EM - Introduce new variables for "Co- ordinate Descent")

Suppose 2 class dataset

D = (xi -> yi ∊ {A , B})i Can we use Gaussian Mixture Model?

DA  = { xi | yi = A }i

Db = { xi | yi = B }i

Get probability distribution

B B B

B B B

A B B B

A A A

A A

PA (x) = P(x | y = A)

PB (x) = P(x | y = B)

Use Baye's Rule

P(x) = PA (x) Pprev (y = A) + Pprev (x) P(y = B) Pprev =Prevelence

P(y = A | x) = [ P(x| y = A) Pprev(y = A) ] / P(x) (here P(x) is given above and P(x |y = A)) = PA (x) )

= [ PA (x) P(y = A) ] / [ (x) Pprev (y = A) + Pprev (x) P(y = B) ]

This is a Bayesian Classifier

here P(y = A) is a "prior" to the classifier

E.g. take PA and PB to be correct but the prior changes?

PA  is the probability that someone is dangerous

PB  is the probability that someone is non-dangerous

Is a person from the college campus dangerous (prior)? -- look at density under the danger, non danger densities

However if you are in a place such as a forest at night, change the prior (more likely to be dangerous)

------------------------------------------------------------------------------------------------------------------------------------

**Aside**-- Are we better off using a SVM or the PA , PB from Baye's Law?

Sometimes it is easier to find the boundary of the data rather than the probability distribution.

Jury still out on which is better but ~ Bayesian Classification is doing better.

Example of this to see which you might use?

Expectation E[ y | x] vs. Probability x |-> P(y | x)

- Bidding for a house.

Expected value ~ 200k -> bid 200k expected to win.

Probability function

0 200k 250k 400k

here- bid of 200k has high probability of succeeding but you could pay less.

How about a different density?

200k 250k

Here a bid of 200K is a lot less probable - must go higher

--------------------------------------------------------------------------------------------------------------------------------------

**Machine Learning in Action** "Spam Filter"

How do we sort "Spam" from "Ham" ? (non spam)

Old way - figure out what is spam, call ISP and tell them to shut off the spam origin. (didn't work)

-- We use a Naive Bayes Classifier.

Consider a message as a sequence of words

x = (x1, x2 , x3 , .... , xn )

P(x) = P(x1, x2 , x3 , .... , xn )

= P(x1) P(x2 | x1) . P(x3 | x1 x2 ). ........ . P(xn | x1, x2 , . . . . , xn-1 )

i.e. the probability of the next word is a contingent of the previous words.

This gives rise to "kth Order Markov Model"

= ∏ P(xi | x1, x2 , . . . . , xi-k) (between i=1 -> n)

1st Order Markov Model

=∏ P(xi | xi-1)

So back to the problem

P(w | u) ? probability of w given previous word u.

=P(w | u) = P(w , u) / P(u)

= [ #(w,u) / N ] / [ #u / N ] N = number of samples (words)

Add all of these pairs into hash tables, one for Spam pairs, one for Ham pairs. When filtering messages, use a threshold i.e. 90% chance it's spam, run it through the hash tables and then send on the filtered messages. Here if new spam comes in, it can be added to the hash tables very easily. (system can keep updating)