http://www.youtube.com/watch?v=nPdP1jBfxzo

http://www.getrobo.com/getrobo/2008/10/keepon-is-now-o.html

Inference in Bayes nets and Naïve Bayes

CS151
David Kauchak
Fall 2010

Some material borrowed from: Sara Owsley Sood and others

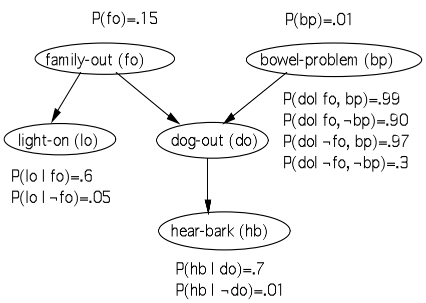
Admin

- Assignment 4 out
- Written 4 and 5
- Grading
- Midterm
- TA office hours

Asking questions about distributions

- We want to be able to ask questions about these probability distributions
- Given *n* variables, a query splits the variables into three sets:
  - query variable(s)
  - known/evidence variables
  - unknown/hidden variables
- P(query | evidence)
  - if we had no hidden variables, we could just multiply all the values in the different CPTs
  - to answer this, we need to sum over the hiden variables!

BN Example



p(fo | hb, lo)?

p(fo | hb, lo)

$$p(fo \mid hb,lo) = \frac{p(fo,hb,lo)}{p(hb,lo)}$$

Evidence: HB, LO
Query: FO
Hidden: BP, DO

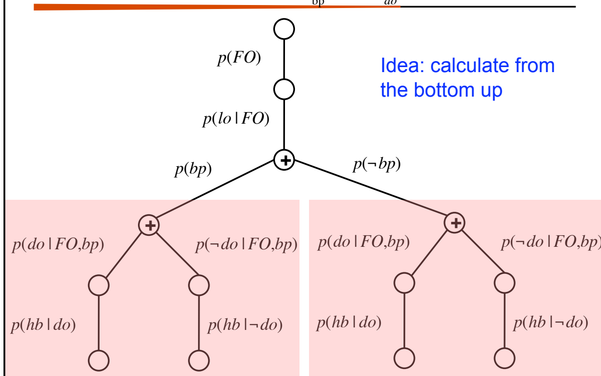
$$p(FO \mid hb,lo) = \alpha\, p(FO,hb,lo)$$

$$= \alpha \sum_{bp}\sum_{do} p(FO,hb,lo,bp,do)$$

$$= \alpha \sum_{bp}\sum_{do} p(FO)p(bp)p(lo \mid FO)p(do \mid FO,bp)p(hb \mid do)$$

$$= \alpha\, p(FO)p(lo \mid FO)\sum_{bp} p(bp)\sum_{do} p(do \mid FO,bp)p(hb \mid do)$$

$$p(FO \mid hb,lo) = \alpha\, p(FO)p(lo \mid FO)\sum_{bp} p(bp)\sum_{do} p(do \mid FO,bp)p(hb \mid do)$$



Idea: calculate from the bottom up

Variable elimination

- Avoids repeated computation
- Break the calculation into *factors*
  - each factor involves some (or all) of the variables
  - factors represent the values for the possible combinations of the variables
  - Initially, these values come straight from the conditional probability tables

$$\underbrace{p(fo)}_{f_1(fo)}\underbrace{p(lo \mid fo)}_{f_2(lo,\ fo)}\sum_{bp}\underbrace{p(bp)}_{f_3(bp)}\sum_{do}\underbrace{p(do \mid fo,bp)}_{f_4(do,fo,bp)}\underbrace{p(hb \mid do)}_{f_5(hb,do)}$$

$f_1(fo)$: $\begin{bmatrix} 0.15 \end{bmatrix}$

$f_2(lo, fo)$: $\begin{bmatrix} 0.6 \end{bmatrix}$

$f_3(bp)$:

bp	
T	0.01
F	0.99

$f_4(do,fo,bp)$:

do	fo	bp	
T	T	T	0.99
T	T	F	0.90
F	T	T	0.01
F	T	F	0.10

$f_5(hb,do)$:

hb	do	
T	T	0.7
T	F	0.01

Variable elimination

- What is the size of the factor's matrix dependent on?
  - the number of hidden variables, m
  - 2^m
  - depending on how you treat query variables, they might also factor in here

$$\underbrace{p(fo)}_{f_1(fo)}\underbrace{p(lo \mid fo)}_{f_2(lo,\, fo)}\sum_{bp}\underbrace{p(bp)}_{f_3(bp)}\sum_{do}\underbrace{p(do \mid fo,bp)}_{f_4(do,fo,bp)}\underbrace{p(hb \mid do)}_{f_5(hb,do)}$$

$$\begin{bmatrix} 0.15 \end{bmatrix} \quad \begin{bmatrix} 0.6 \end{bmatrix} \quad \begin{pmatrix} bp & \\ T & 0.01 \\ F & 0.99 \end{pmatrix} \quad \begin{pmatrix} do & fo & bp & \\ T & T & T & 0.99 \\ T & T & F & 0.90 \\ F & T & T & 0.01 \\ F & T & F & 0.10 \end{pmatrix} \quad \begin{pmatrix} hb & do & \\ T & T & 0.7 \\ T & F & 0.01 \end{pmatrix}$$

Variable elimination

$$f_1(fo)f_2(lo,fo)\sum_{bp} f_3(bp)\sum_{do} f_4(do,fo,bp)f_5(hb,do)$$

$$\begin{bmatrix} 0.15 \end{bmatrix} \quad \begin{bmatrix} 0.6 \end{bmatrix} \quad \begin{pmatrix} bp & \\ T & 0.01 \\ F & 0.99 \end{pmatrix} \quad \begin{pmatrix} do & fo & bp & \\ T & T & T & 0.99 \\ T & T & F & 0.90 \\ F & T & T & 0.01 \\ F & T & F & 0.10 \end{pmatrix} \quad \begin{pmatrix} hb & do & \\ T & T & 0.7 \\ T & F & 0.01 \end{pmatrix}$$

- Solve this from right to left using two operations:
  - pointwise product of factors
  - summing out a variable

Pointwise product

$$f_1(x_1,\ldots,x_n,y_1,\ldots,y_m)f_2(y_1,,\ldots,y_n,z_1,\ldots,z_p) =$$
$$f_3(x_1,,\ldots,x_n,y_1,,\ldots,y_n,z_1,\ldots,z_p)$$

- When we take the product of two factors, we have three sets of variables
  - $x_1,\ldots,x_n$: those unique to f_1
  - $z_1,\ldots,z_p$: those unique to f_2
  - $y_1,\ldots,y_n$: those shared between the two
- The result is a *new* factor over the union of the variables

Pointwise product

$$f_4(do,fo,bp)f_5(hb,do) = f_6(do,fo,bp,hp)$$

$$\begin{pmatrix} do & fo & bp & \\ T & T & T & 0.99 \\ T & T & F & 0.90 \\ F & T & T & 0.01 \\ F & T & F & 0.10 \end{pmatrix} \quad \begin{pmatrix} hb & do & \\ T & T & 0.7 \\ T & F & 0.01 \end{pmatrix} \quad \begin{pmatrix} do & fo & bp & hb & \\ \boxed{T \; T \; T \; T} & & & & \\ T & T & F & T & \\ F & T & T & T & \\ F & T & F & T & \end{pmatrix}$$

Pointwise product

$$f_4(do,fo,bp)\,f_5(hb,do) = f_6(do,fo,bp,hp)$$

do	fo	bp	
T	T	T	0.99
T	T	F	0.90
F	T	T	0.01
F	T	F	0.10

hb	do	
T	T	0.7
T	F	0.01

do	fo	bp	hb	
T	T	T	T	0.7*0.99
T	T	F	T	
F	T	T	T	
F	T	F	T	

Pointwise product

$$f_4(do,fo,bp)\,f_5(hb,do) = f_6(do,fo,bp,hp)$$

do	fo	bp	
T	T	T	0.99
T	T	F	0.90
F	T	T	0.01
F	T	F	0.10

hb	do	
T	T	0.7
T	F	0.01

do	fo	bp	hb	
T	T	T	T	0.7*0.99
T	T	F	T	0.7*0.90
F	T	T	T	
F	T	F	T	

Pointwise product

$$f_4(do,fo,bp)\,f_5(hb,do) = f_6(do,fo,bp,hp)$$

do	fo	bp	
T	T	T	0.99
T	T	F	0.90
F	T	T	0.01
F	T	F	0.10

hb	do	
T	T	0.7
T	F	0.01

do	fo	bp	hb	
T	T	T	T	0.7*0.99
T	T	F	T	0.7*0.90
F	T	T	T	0.01*0.01
F	T	F	T	0.01*0.10

In this case the size of the factor didn't increase, but in general, it can

Pointwise Product

A	B	f_1(A,B)	B	C	f_2(B,C)	A	B	C	f_3(A,B,C)
T	T	.3	T	T	.2	T	T	T	0.06
T	F	.7	T	F	.8	T	T	F	0.24
F	T	.9	F	T	.6	T	F	T	0.42
F	F	.1	F	F	.4	T	F	F	0.28
						F	T	T	0.18
						F	T	F	0.72
						F	F	T	0.06
						F	F	F	0.04

Summing out a variable

$$f_1(fo)f_2(lo,fo)\sum_{bp} f_3(bp)\boxed{\sum_{do} f_6(do,fo,bp,hb)}$$

$\begin{bmatrix} 0.15 \end{bmatrix}$ $\begin{bmatrix} 0.6 \end{bmatrix}$

bp	
T	0.01
F	0.99

do	fo	bp	hb	
T	T	T	T	0.7*0.99
T	T	F	T	0.7*0.90
F	T	T	T	0.01*0.01
F	T	F	T	0.01*0.10

- Produces a new factor with one less variable
- Reduces the size of the table by a factor of the number of possible values for the variable (for binary 2)

Summing out a variable

$$\sum_{do} f_6(do,fo,bp,hb) = f_6(do,fo,bp,hp) + f_6(\neg do,fo,bp,hp)$$
$$= f_7(fo,bp,hp)$$

do	fo	bp	hb	
T	T	T	T	0.7*0.99
T	T	F	T	0.7*0.90
F	T	T	T	0.01*0.01
F	T	F	T	0.01*0.10

fo	bp	hb	
T	T	T	0.7*0.99 + 0.01*0.01
T	F	T	0.7*0.90 + 0.01*0.10

How do we sum out a variable?

Variable Elimination

$$f_1(fo)f_2(lo,fo)\sum_{bp} f_3(bp)f_7(fo,bp,hb)$$

$\begin{bmatrix} 0.15 \end{bmatrix}$ $\begin{bmatrix} 0.6 \end{bmatrix}$

bp	
T	0.01
F	0.99

fo	bp	hb	
T	T	T	0.7*0.99 + 0.01*0.01
T	F	T	0.7*0.90 + 0.01*0.10

$f_1(fo)f_2(lo,fo)\sum_{bp} f_8(fo,bp,hb)$ product

$f_1(fo)f_2(lo,fo)f_9(fo,hb)$ sum

$\vdots$

Variable ordering

$$f_1(fo)f_2(lo,fo)\sum_{bp} f_3(bp)\sum_{do} f_4(do,fo,bp)f_5(hb,do)$$

$$=$$

$$f_1(fo)f_2(lo,fo)\sum_{do} f_5(hb,do)\sum_{bp} f_3(bp)f_4(do,fo,bp)$$

- The complexity depends on which order we sum out the variables

Variable ordering

$$\sum_a \sum_b f_1(a,b) f_2(b,c) \quad \text{vs} \quad \sum_b f_2(b,c) \sum_a f_1(a,b)$$

$$\underbrace{\sum_a \sum_b f_3(a,b,c)} \qquad \underbrace{\sum_b f_2(b,c) f_3(b)}$$

A factor containing 3 variables

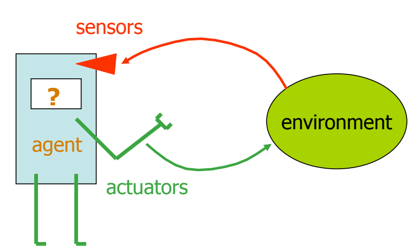
A factor containing only 2 variables

Runtime

- In general, the run-time of the variable elimination algorithm is dependent on the largest factor created
- Figuring out the optimal variable ordering is intractable
- Some heuristics have been used
  - pick the merger greedily

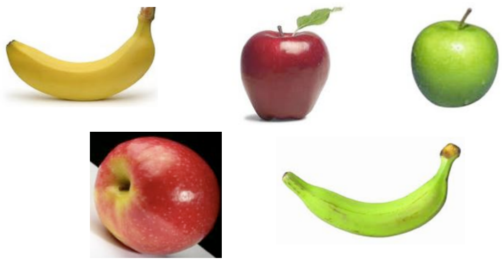
Learning from Data

Learning



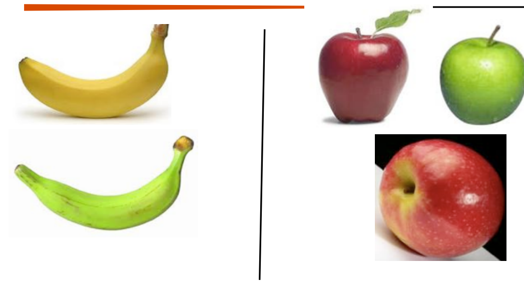
As an agent interacts with the world, it should learn about it's environment

Lots of different learning problems



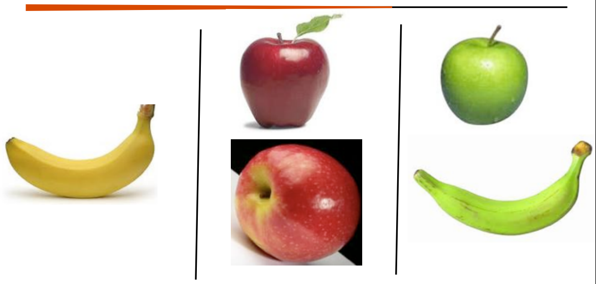
Unsupervised learning: put these into groups

Lots of different learning problems



Unsupervised learning: put these into groups

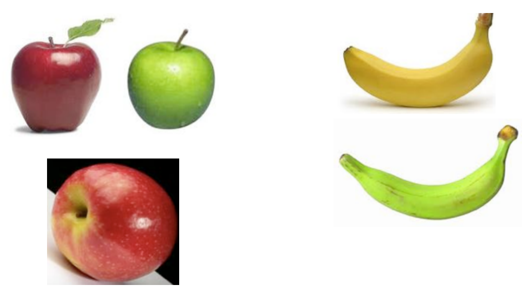
Lots of different learning problems



No explicit labels/categories specified

Unsupervised learning: put these into groups

Lots of learning problems



APPLES **BANANAS**

Supervised learning: given labeled data

Lots of learning problems

- Given labeled examples, learn to label unlabeled examples

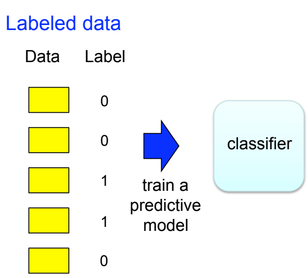
APPLE or BANANA?

Supervised learning: learn to classify unlabeled

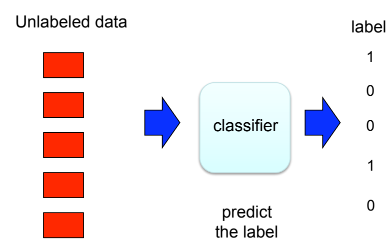
Lots of learning problems

- Many others
  - semi-supervised learning: some labeled data and some unlabeled data
  - active learning: unlabeled data, but we can pick some examples to be labeled
  - reinforcement learning: maximize a *cumulative* reward. Learn to drive a car, reward = not crashing
- and variations
  - online vs. offline learning: do we have access to all of the data or do we have to learn as we go
  - classification vs. regression: are we predicting between a finite set or are we predicting a score/value

Supervised classification: training



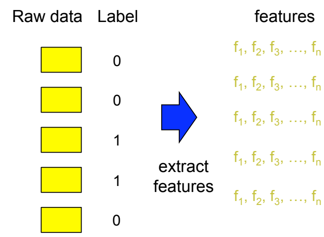
Supervised learning: testing/classifying



Some example

- image classification
  - does the image contain a person? apple? banana?
- text classification
  - is this a good/bad review?
  - is this article about sports or politics?
  - is this e-mail spam?
- character recognition
  - is this set of scribbles an ‘a’, ‘b’, ‘c’, …
- credit card transactions
  - fraud or not?
- audio classification
  - hit or not?
  - jazz, pop, blues, rap, …
- Tons of problems!!!

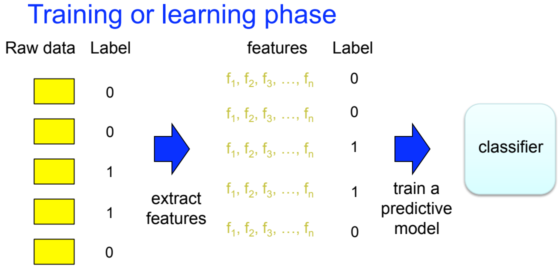
Features



- We’re given “raw data”, e.g. text documents, images, audio, …
- Need to extract “features” from these (or to think of it another way, we somehow need to represent these things)
- What might be features for: text, images, audio?

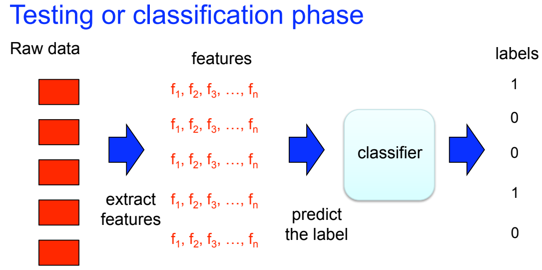
Feature based classification

Training or learning phase



Feature based classification

Testing or classification phase



Bayesian Classification

We represent a data item based on the features:

$$D = \langle f_1, f_2, \ldots, f_n \rangle$$

Training

a: $p(a \mid D) = p(a \mid f_1, f_2, \ldots, f_n)$

b: $p(b \mid D) = p(b \mid f_1, f_2, \ldots, f_n)$

$\rightarrow P(Label \mid f_1, f_2, \ldots, f_n)$

For each label/class, **learn** a probability distribution based on the features

Bayesian Classification

We represent a data item based on the features:

$$D = \langle f_1, f_2, \ldots, f_n \rangle$$

Classifying

$$label = \underset{l \in Labels}{\operatorname{argmax}} P(l \mid f_1, f_2, \ldots, f_n)$$

Given an *new* example, classify it as the label with the largest conditional probability

Bayes rule for classification

conditional (posterior) probability

prior probability

$$P(Label \mid Data) = \frac{P(D \mid C)P(C)}{P(D)}$$

Bayesian Classifiers

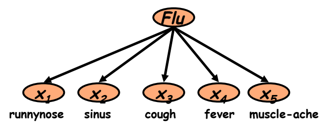
$$label = \underset{l \in Labels}{\operatorname{argmax}} P(l \mid f_1, f_2, \ldots, f_n) \longleftarrow$$ different distributions for different labels

$$= \underset{l \in Labels}{\operatorname{argmax}} \frac{P(f_1, f_2, \ldots, f_n \mid l)P(l)}{P(f_1, f_2, \ldots, f_n)}$$ Bayes rule

$$= \underset{l \in Labels}{\operatorname{argmax}} P(f_1, f_2, \ldots, f_n \mid l)P(l)$$

two models to learn for each label/class

The Naive Bayes Classifier



Conditional Independence Assumption: features are independent of eachother given the class:

$$P(x_1,\ldots,x_n \mid l) = P(x_1 \mid l)P(x_2 \mid l)\cdots P(x_n \mid l)$$

$$label = \operatorname*{argmax}_{l \in Labels} P(f_1 \mid l)P(f_2 \mid l)\ldots p(f_n \mid l)P(l)$$

Estimating parameters

- I flip a coin 1000 times, how would you estimate the probability of heads?
- I roll a 6-sided die 1000 times, how you estimate the probability of getting a '6'?

For us:

$$label = \operatorname*{argmax}_{l \in Labels} P(f_1 \mid l)P(f_2 \mid l)\ldots p(f_n \mid l)P(l)$$

Ideas?

Maximum likelihood estimates

$$\hat{P}(l) = \frac{N(l)}{N} \qquad \frac{\text{number of items with label}}{\text{total number of items}}$$

$$\hat{P}(f_i \mid l) = \frac{N(f_i, l)}{N(l)} \qquad \frac{\text{number of items with the label with feature}}{\text{number of items with label}}$$

Any problems with this approach?

Problem with Max Likelihood

- What if we have seen no training cases where patient had no flu and muscle aches?

$$\hat{P}(f_{mu} \mid nf) = \frac{N(f_{mu}, nf)}{N(nf)} = 0$$

- Zero probabilities cannot be conditioned away, no matter the other evidence!

$$label = \operatorname{arg\,max}_{l \in Labels} \hat{P}(l) \prod_i \hat{P}(f_i \mid l)$$

Smoothing to Avoid Overfitting

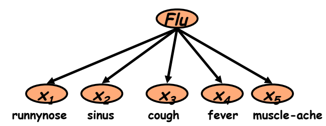
Make every event a little probable…

$$\hat{P}(f_i \mid l) = \frac{N(f_i, l) + \lambda}{N(l) + k\lambda}$$

# of features

Unseen features

- Note that this is different from coming in with a feature we've never seen before (in any of the classes)
  - For example, "bloating"



Naïve Bayes Text Classification

- Features: word occurring in a document (though others could be used…)

$$label = \underset{l \in Labels}{\operatorname{argmax}} P(word_1 \mid l) P(word_2 \mid l) \ldots p(word_n \mid l) P(l)$$

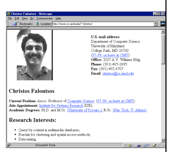
- Does the Naïve Bayes assumption hold?
  - Are word occurrences independent given the label?
- We'll look at a few application for this homework
  - sentiment analysis: positive vs. negative reviews
  - category classifiction

Classification evaluation?

- Accuracy
  - num correct / total
- Class specific measures
  - Precision
    - num correct with class A / num predicted class A
  - Recall
    - num correct with class A / num with class A
  - F1-measure
    - 2 * (precision * recall) / (precision + recall)
- Why have these class specific measures?

WebKB Experiment (1998)

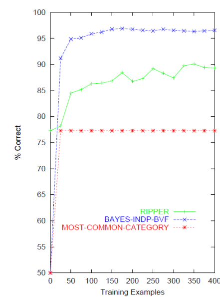
- Classify webpages from CS departments into:
  - student, faculty, course,project
- Train on ~5,000 hand-labeled web pages
  - Cornell, Washington, U.Texas, Wisconsin
- Crawl and classify a new site (CMU)



- Results:

	Student	Faculty	Person	Project	Course	Departmt
Extracted	180	66	246	99	28	1
Correct	130	28	194	72	25	1
Accuracy:	72%	42%	79%	73%	89%	100%

Naive Bayes on spam email



http://www.cnbc.cmu.edu/~jp/research/email.paper.pdf

SpamAssassin

- Naive Bayes has found a home in spam filtering
  - Paul Graham's *A Plan for Spam*
    - A mutant with more mutant offspring...
  - Naive Bayes-like classifier with weird parameter estimation
  - Widely used in spam filters
  - But also many other things: black hole lists, etc.
- Many email topic filters also use NB classifiers

NB: The good and the bad

- Good
  - Easy to understand
  - Fast to train
  - Reasonable performance
- Bad
  - We can do better
  - Independence assumptions are rarely true
  - Smoothing is challenging
  - Feature selection is usually required