Probabilistic and Bayesian Learning

Ronald J. Williams

CSG220

Spring 2007

Note to other teachers and users of these slides. Andrew would be delighted if you found this source material useful in giving your own lectures. Feel free to use these slides verbatim, or to modify them to fit your own needs. PowerPoint originals are available. If you make use of a significant portion of these slides in your own lecture, please include this message, or the following link to the source repository of Andrew's tutorials: http://www.cs.cmu.edu/~awm/tutorials . Comments and corrections gratefully received.

Containing many slides adapted from the Andrew Moore tutorial "Probabilistic and Bayesian Analytics"

Probability

- The world is a very uncertain place
- 30 years of Artificial Intelligence and Database research danced around this fact
- And then a few AI researchers decided to use some ideas from the eighteenth century

What we're going to do

- We will review the fundamentals of probability.
- It's really going to be worth it
- In this lecture, you'll see an example of probabilistic analysis in action: Bayes Classifiers

Discrete Random Variables

- E is a Boolean-valued random variable if E denotes an event, and there is some degree of uncertainty as to whether E occurs.
- Examples
- E = The US president in 2023 will be male
- E = You wake up tomorrow with a headache
- E = You have Ebola
- E = (Outlook = sunny) and (Wind = strong)

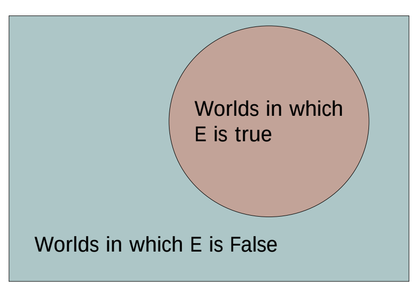
Probabilities

- We write P(E) as “the fraction of possible worlds in which E is true”
- We could at this point spend 2 hours on the philosophy of this.
- But we won’t.

Visualizing E

Event space of all possible worlds

Its area is 1



P(E) = Area of brown circle

The Axioms of Probability

- $0 <= P(E) <= 1$
- $P(True) = 1$
- $P(False) = 0$
- $P(E_1 \text{ or } E_2) = P(E_1) + P(E_2) - P(E_1 \text{ and } E_2)$

These Axioms are Not to be Trifled With

- There have been attempts to do different methodologies for uncertainty
  - Fuzzy Logic
  - Three-valued logic
  - Dempster-Shafer
  - Non-monotonic reasoning
- But the axioms of probability are the only system with this property:

  If you gamble using them you can't be unfairly exploited by an opponent using some other system [di Finetti 1931]

Theorems from the Axioms

Easy consequences of the axioms:

- $P(\sim E) = 1 - P(E)$
- $P(E_1) = P(E_1 \wedge E_2) + P(E_1 \wedge \sim E_2)$

Multivalued Random Variables

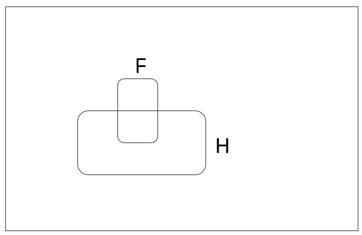
- Suppose A can take on any of several values
- A is a *random variable with arity k* if it can take on exactly one value out of $\{v_1, v_2, .. v_k\}$
- Thus

$$P(A = v_i \wedge A = v_j) = 0 \text{ if } i \neq j$$

$$P(A = v_1 \vee A = v_2 \vee \ldots \vee A = v_k) = 1$$

Conditional Probability

- $P(E_1|E_2)$ = Fraction of worlds in which E_2 is true that also have E_1 true

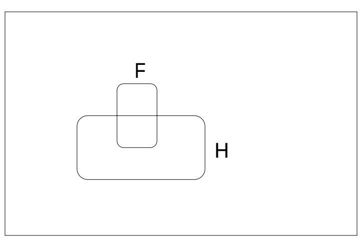


H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

"Headaches are rare and flu is rarer, but if you're coming down with flu there's a 50-50 chance you'll have a headache."

Conditional Probability



H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

P(H|F) = Fraction of flu-inflicted worlds in which you have a headache

$$= \frac{\#\text{worlds with flu and headache}}{\#\text{worlds with flu}}$$

$$= \frac{\text{Area of "H and F" region}}{\text{Area of "F" region}}$$

$$= \frac{P(H \wedge F)}{P(F)}$$

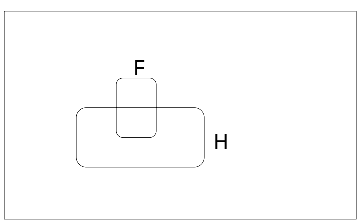
Definition of Conditional Probability

$$P(E_1|E_2) = \frac{P(E_1 \wedge E_2)}{P(E_2)}$$

Corollary: The Chain Rule

$$P(E_1 \wedge E_2) = P(E_1|E_2)\, P(E_2)$$

Probabilistic Inference



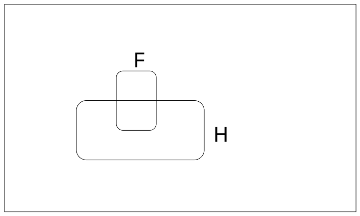
H = "Have a headache"
F = "Coming down with Flu"

$P(H) = 1/10$
$P(F) = 1/40$
$P(H|F) = 1/2$

One day you wake up with a headache. You think: "Drat! 50% of flus are associated with headaches so I must have a 50-50 chance of coming down with flu"

Is this reasoning good?

Probabilistic Inference



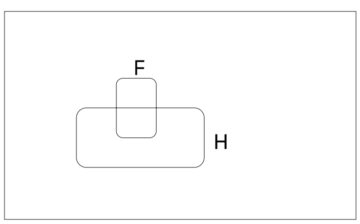
H = “Have a headache”
F = “Coming down with Flu”

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

P(F ^ H) = ...

P(F|H) = ...

Probabilistic Inference



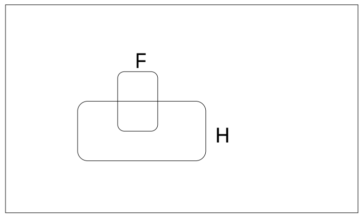
H = “Have a headache”
F = “Coming down with Flu”

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

P(F ^ H) = P(H ^ F) = P(H|F) P(F) = (1/2)*(1/40) = 1/80

P(F|H) = ...

Probabilistic Inference



H = “Have a headache”
F = “Coming down with Flu”

$P(H) = 1/10$
$P(F) = 1/40$
$P(H|F) = 1/2$

$$P(F \wedge H) = P(H \wedge F) = P(H|F)\,P(F) = (1/2)*(1/40) = 1/80$$

$$P(F|H) = \frac{P(F \wedge H)}{P(H)} = \frac{1/80}{1/10} = 1/8$$

What we just did...

$$P(E_2|E_1) = \frac{P(E_1 \wedge E_2)}{P(E_1)} = \frac{P(E_1|E_2)\,P(E_2)}{P(E_1)}$$

This is Bayes’ Rule

Bayes, Thomas (1763) An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London,* **53:370-418**

More General Forms of Bayes Rule

$$P(E|F) = \frac{P(F|E)P(E)}{P(F|E)P(E) + P(F|\sim E)P(\sim E)}$$

$$P(E|F \wedge G) = \frac{P(F|E \wedge G)P(E \wedge G)}{P(F \wedge G)}$$

More General Forms of Bayes Rule

$$P(A=v_i|F) = \frac{P(F|A=v_i)P(A=v_i)}{\sum_{k=1}^{n_A} P(F|A=v_k)P(A=v_k)}$$

Useful Easy-to-prove facts

$$P(E \mid F) + P(\sim E \mid F) = 1$$

$$\sum_{k=1}^{n_A} P(A = v_k \mid F) = 1$$

The Joint Distribution

- If A_1, A_2, ... , A_n are multivalued random variables,

$$P(A_1, A_2, \ldots, A_n)$$

means the function assigning to any v_1, v_2, ... , v_n the probability

$$P(A_1 = v_1 \wedge A_2 = v_2 \wedge \ldots \wedge A_n = v_n)$$

Conditional Distributions

- Suppose we have a joint distribution over the *n+m* multivalued random variables A_1, A_2, ... , A_n, B_1, B_2, ... , B_m.

Conditional Distributions

- Then

$$P(A_1, A_2, \ldots, A_n \mid B_1, B_2, \ldots, B_m)$$

means the function assigning to any u_1, u_2, ... , u_n , v_1, v_2, ..., v_m the conditional probability

$$P(A_1 = u_1 \wedge \ldots \wedge A_n = u_n \mid B_1 = v_1 \wedge \ldots \wedge B_m = v_m)$$

Bayesian Hypothesis Learning

- D = training data
- H = hypothesis (treated as random variable)
- P(H) = prior distribution over hypotheses
  - formalizes inductive bias
- P(H|D) = posterior distribution
  - after seeing the training data
- Then

$$P(H \mid D) = \frac{P(D \mid H)P(H)}{P(D)}$$

Bayesian Hypothesis Learning

- D = training data
- H = hypothesis (treated as random variable)
- P(H) = prior distribution over hypotheses
  - formalizes inductive bias
- P(H|D) = posterior distribution
  - after seeing the training data
- Then

$$P(H \mid D) = \frac{\boxed{P(D \mid H)}P(H)}{P(D)}$$

likelihood of the data

Bayesian Hypothesis Learning

- D = training data
- H = hypothesis (treated as random variable)
- P(H) = prior distribution over hypotheses
  - formalizes inductive bias
- P(H|D) = posterior distribution
  - after seeing the training data
- Then

$$P(H \mid D) = \frac{P(D \mid H) P(H)}{\boxed{P(D)}}$$

Fixed for any given set of training data – can ignore and treat as a normalizing constant

Bayesian Hypothesis Learning

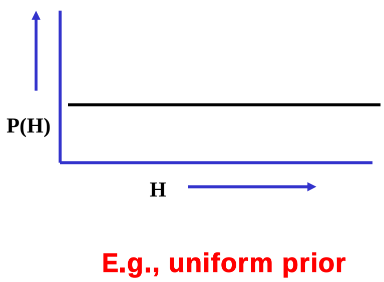
- Given data d, want hypothesis h
- Use

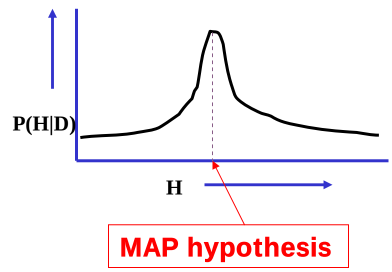
$$P(H = h \mid D = d) \propto P(D = d \mid H = h) P(H = h)$$

- Maximum *a posteriori* (MAP) hypothesis:
  - h maximizing P(H=h|D=d)
- Maximum likelihood (ML) hypothesis:
  - h maximizing P(D=d|H=h)
- If P(H) is uniform ("flat prior"), they're the same

Bayesian Hypothesis Learning

- *a priori* distribution
  - before seeing the data
- *a posteriori* distribution
  - after seeing the data





The Joint Distribution

Example: Boolean variables A, B, C

Recipe for making a joint distribution of M variables:

The Joint Distribution

Example: Boolean variables A, B, C

Recipe for making a joint distribution of M variables:

1. Make a truth table listing all combinations of values of your variables (if there are M Boolean variables then the table will have 2^M rows).

A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

The Joint Distribution

Example: Boolean variables A, B, C

Recipe for making a joint distribution of M variables:

1. Make a truth table listing all combinations of values of your variables (if there are M Boolean variables then the table will have 2^M rows).
2. For each combination of values, say how probable it is.

A	B	C	Prob
0	0	0	0.30
0	0	1	0.05
0	1	0	0.10
0	1	1	0.05
1	0	0	0.05
1	0	1	0.10
1	1	0	0.25
1	1	1	0.10

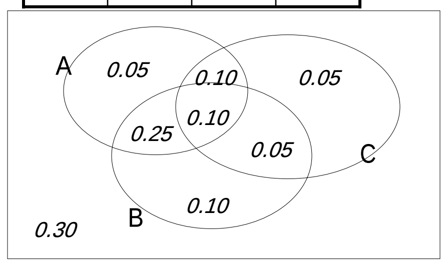
The Joint Distribution

Example: Boolean variables A, B, C

Recipe for making a joint distribution of M variables:

1. Make a truth table listing all combinations of values of your variables (if there are M Boolean variables then the table will have 2^M rows).
2. For each combination of values, say how probable it is.
3. If you subscribe to the axioms of probability, those numbers must sum to 1.

A	B	C	Prob
0	0	0	0.30
0	0	1	0.05
0	1	0	0.10
0	1	1	0.05
1	0	0	0.05
1	0	1	0.10
1	1	0	0.25
1	1	1	0.10



Using the Joint

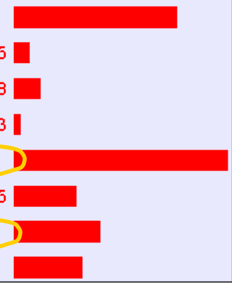
gender	hours_worked	wealth	
Female	v0:40.5-	poor	0.253122
		rich	0.0245895
	v1:40.5+	poor	0.0421768
		rich	0.0116293
Male	v0:40.5-	poor	0.331313
		rich	0.0971295
	v1:40.5+	poor	0.134106
		rich	0.105933

Once you have the JD you can ask for the probability of any logical expression involving your attribute

$$P(E) = \sum_{\text{rows matching } E} P(\text{row})$$

Using the Joint

gender	hours_worked	wealth	
Female	v0:40.5-	poor	0.253122
		rich	0.0245895
	v1:40.5+	poor	0.0421768
		rich	0.0116293
Male	v0:40.5-	poor	0.331313
		rich	0.0971295
	v1:40.5+	poor	0.134106
		rich	0.105933

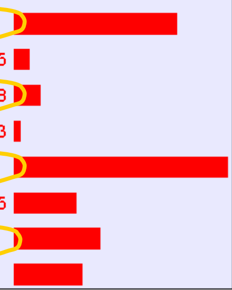


P(Poor Male) = 0.4654

$$P(E) = \sum_{\text{rows matching } E} P(\text{row})$$

Using the Joint

gender	hours_worked	wealth	
Female	v0:40.5-	poor	0.253122
		rich	0.0245895
	v1:40.5+	poor	0.0421768
		rich	0.0116293
Male	v0:40.5-	poor	0.331313
		rich	0.0971295
	v1:40.5+	poor	0.134106
		rich	0.105933



P(Poor) = 0.7604

$$P(E) = \sum_{\text{rows matching } E} P(\text{row})$$

Inference with the Joint

gender	hours_worked	wealth	
Female	v0:40.5-	poor	0.253122
		rich	0.0245895
	v1:40.5+	poor	0.0421768
		rich	0.0116293
Male	v0:40.5-	poor	0.331313
		rich	0.0971295
	v1:40.5+	poor	0.134106
		rich	0.105933

$$P(E_1 \mid E_2) = \frac{P(E_1 \wedge E_2)}{P(E_2)} = \frac{\sum_{\text{rows matching } E_1 \text{ and } E_2} P(\text{row})}{\sum_{\text{rows matching } E_2} P(\text{row})}$$

Inference with the Joint

gender	hours_worked	wealth	
Female	v0:40.5-	poor	0.253122
		rich	0.0245895
	v1:40.5+	poor	0.0421768
		rich	0.0116293
Male	v0:40.5-	poor	0.331313
		rich	0.0971295
	v1:40.5+	poor	0.134106
		rich	0.105933

$$P(E_1 \mid E_2) = \frac{P(E_1 \wedge E_2)}{P(E_2)} = \frac{\sum_{\text{rows matching } E_1 \text{ and } E_2} P(\text{row})}{\sum_{\text{rows matching } E_2} P(\text{row})}$$

P(Male | Poor) = 0.4654 / 0.7604 = 0.612

Inference is a big deal

- I've got this evidence. What's the chance that this conclusion is true?
  - I've got a sore neck: how likely am I to have meningitis?
  - I see my lights are out and it's 9pm. What's the chance my spouse is already asleep?

Inference is a big deal

- I've got this evidence. What's the chance that this conclusion is true?
  - I've got a sore neck: how likely am I to have meningitis?
  - I see my lights are out and it's 9pm. What's the chance my spouse is already asleep?

- There's a thriving set of industries growing based around Bayesian Inference. Highlights are: Medicine, Pharma, Help Desk Support, Engine Fault Diagnosis

Where do Joint Distributions come from?

- Idea One: Expert Humans
- Idea Two: Simpler probabilistic facts and some algebra

Example: Suppose you knew

P(A) = 0.7

P(B|A) = 0.2

P(B|~A) = 0.1

P(C|A^B) = 0.1

P(C|A^~B) = 0.8

P(C|~A^B) = 0.3

P(C|~A^~B) = 0.1

Then you can automatically compute the JD using the chain rule

P(A=x ^ B=y ^ C=z) =
P(C=z|A=x^ B=y) P(B=y|A=x) P(A=x)

Essential idea behind inference in Bayesian networks

Where do Joint Distributions come from?

- Idea Three: Learn them from data!

Prepare to see one of the most impressive learning algorithms you'll come across in the entire course....

Learning a joint distribution

Build a JD table for your attributes in which the probabilities are unspecified

A	B	C	Prob
0	0	0	?
0	0	1	?
0	1	0	?
0	1	1	?
1	0	0	?
1	0	1	?
1	1	0	?
1	1	1	?

The fill in each row with

$$\hat{P}(\text{row}) = \frac{\text{records matching row}}{\text{total number of records}}$$

A	B	C	Prob
0	0	0	0.30
0	0	1	0.05
0	1	0	0.10
0	1	1	0.05
1	0	0	0.05
1	0	1	0.10
1	1	0	**0.25**
1	1	1	0.10

Fraction of all records in which A and B are True but C is False

Example of Learning a Joint

- This Joint was obtained by learning from three attributes in the UCI "Adult" Census Database [Kohavi 1995]

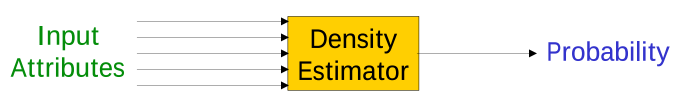
gender	hours_worked	wealth	
Female	v0:40.5-	poor	0.253122
		rich	0.0245895
	v1:40.5+	poor	0.0421768
		rich	0.0116293
Male	v0:40.5-	poor	0.331313
		rich	0.0971295
	v1:40.5+	poor	0.134106
		rich	0.105933

Where are we?

- We have recalled the fundamentals of probability
- We have become content with what JDs are and how to use them
- And we even know how to learn JDs from data.

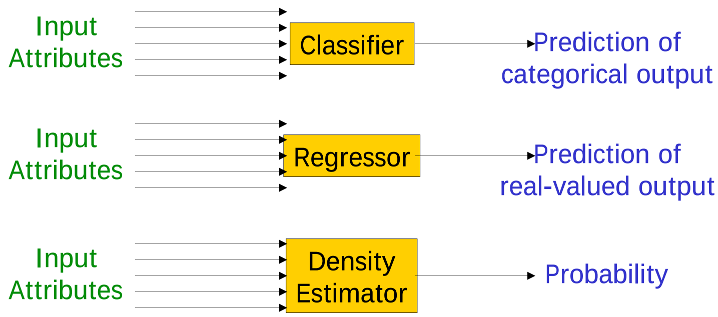
Density Estimation

- Our Joint Distribution learner is our first example of something called Density Estimation
- A Density Estimator learns a mapping from a set of attributes to a Probability



Density Estimation

- Compare it against the two other major kinds of models:



Summary: The Good News

- We have a way to learn a Density Estimator from data.
- Density estimators can do many good things…
  - Can sort the records by probability, and thus spot weird records (anomaly detection)
  - Can do inference: $P(E_1|E_2)$

    Automatic Doctor / Help Desk etc
  - Ingredient for Bayes Classifiers (see later)

Summary: The Bad News

- Density estimation by directly learning the joint
  - is trivial and mindless
  - requires an amount of training data exponential in the number of attributes
- Fortunately there are alternatives ...

PlayTennis Example

- Want joint P(O, T, H, W, PT), where
  Outlook values are {sunny, overcast, rain}
  Temperature values are {hot, mild, cool}
  Humidity values are {high, normal}
  Wind values are {weak, strong}
  PlayTennis values are {yes, no}

PlayTennis Example: Directly Learning the Joint

- Need total of 3*3*2*2*2 = 72 probabilities (71 independent numbers since they sum to 1)
- Have 14 training examples
- Simple-minded estimation of the joint would assign probability 1/14 to the training examples and probability 0 to the remaining 56 possible combinations

Naïve Density Estimation

The problem with the Joint Estimator is that it just mirrors the training data.

It has no possibility of generalizing reasonably to unseen data.

The naïve model generalizes strongly:

Assume that each attribute is distributed independently of any of the other attributes.

Independent Events

- Let E_1 and E_2 be events. Then E_1 and E_2 are independent if and only if

$$P(E_1|E_2) = P(E_1)$$

- Means knowing that E_2 is true has no effect on the probability that E_1 is true.
- "E_1 and E_2 are independent" is often denoted by

$$E_1 \perp E_2$$

Independence Theorems

- Assume E_1 and E_2 are independent.
- Then
  - $P(E_1 \wedge E_2) = P(E_1)\, P(E_2)$
  - $P(E_2|E_1) = P(E_2)$
  - $P(\sim E_1|E_2) = P(\sim E_1)$
  - $P(E_1|\sim E_2) = P(E_1)$

Multivalued Independence

For multivalued Random Variables A_1, ..., A_n, B_1, ..., B_m,

$$\{A_1,\ldots,A_n\} \perp \{B_1,\ldots,B_m\}$$

if and only if

$$\forall u_1,\cdots,u_n,v_1,\ldots,v_m$$
$$P(A_1=u_1 \wedge \ldots \wedge A_n=u_n \mid B_1=v_1 \wedge \ldots \wedge B_m=v_m)$$
$$= P(A_1=u_1 \wedge \ldots \wedge A_n=u_n)$$

Definition: Mutual Independence

Set of random variables $\{A_1, \ldots, A_n\}$ satisfying

$$A_i \perp \{A_1,\ldots,A_{i-1},A_{i+1},\ldots,A_n\} \quad \forall i$$

In this case, the joint satisfies

$$P(A_1,\ldots A_n) = \prod_{i=1}^{n} P(A_i)$$

Back to Naïve Density Estimation

- Let x[i] denote the i'th field of record x:
- Naïve DE assumes *x[i]* is independent of *{x[1],x[2],..x[i-1], x[i+1],...x[M]}*
- Example:
  - Suppose that each record is generated by randomly rolling a green die and a red die
    - Dataset 1: A = red value, B = green value
    - Dataset 2: A = red value, B = sum of values
    - Dataset 3: A = sum of values, B = difference of values
  - Which of these datasets violates the naïve assumption?

Using the Naïve Distribution

- Once you have a Naïve Distribution you can easily compute any row of the joint distribution.
- Suppose *A, B, C* and *D* are mutually independently distributed. What is *P(A^~B^C^~D)*?

Using the Naïve Distribution

- Once you have a Naïve Distribution you can easily compute any row of the joint distribution.
- Suppose A, B, C and D are independently distributed. What is P(A^~B^C^~D)?

= P(A|~B^C^~D) P(~B^C^~D)

= P(A) P(~B^C^~D)

= P(A) P(~B|C^~D) P(C^~D)

= P(A) P(~B) P(C^~D)

= P(A) P(~B) P(C|~D) P(~D)

= P(A) P(~B) P(C) P(~D)

Naïve Distribution General Case

- Suppose *x[1], x[2], ... x[M]* are independently distributed.

$$P(x[1]=u_1, x[2]=u_2, \ldots x[M]=u_M) = \prod_{k=1}^{M} P(x[k]=u_k)$$

- So if we have a Naïve Distribution we can construct any row of the implied Joint Distribution on demand.
- So we can do any inference
- But how do we learn a Naïve Density Estimator?

Learning a Naïve Density Estimator

$$\hat{P}(x[i]=u) = \frac{\#\text{records in which } x[i]=u}{\text{total number of records}}$$

Another trivial learning algorithm!

Contrast

Direct Joint DE	Naïve DE
Can model anything	Can model only very boring distributions
Given 100 records and more than 6 Boolean attributes will screw up badly	Given 100 records and 10,000 multivalued attributes will be fine

Reminder: The Good News

- We have two ways to learn a Density Estimator from data.
- There are many other vastly more impressive Density Estimators (Mixture Models, Bayesian Networks, Density Trees, Kernel Densities and many more)
- Density estimators can do many good things…
  - Anomaly detection
  - Can do inference: $P(E_1|E_2)$ Automatic Doctor / Help Desk etc
  - Ingredient for Bayes Classifiers

Bayes Classifiers

- Let Y be the class (a random variable) and X a random vector of input attributes.
- If we estimate the joint P(X, Y) from training data, given a vector of values x we can classify x by selecting the value of y maximizing P(Y=y | X=x).
- This is all there is to a Bayes classifier.
- Any way of estimating the joint gives rise to a corresponding Bayes classifier.

Bayes Classifiers

Ways of estimating the joint

1. Directly from data: Gives rise to a useless classifier unless we have lots of data. Really just memorization of the data with no real generalization.
2. Make the naïve assumption for P(X, Y): No good either because then P(Y | X) = P(Y) so the result does not depend on the input attributes.
3. Assume conditional independence of attributes given the class. We'll examine this in a moment. This yields the *naïve Bayes* classifier.

How to build a Bayes Classifier

- Assume you want to predict output Y which has arity n_Y and values v_1, v_2, … v_{ny}.
- Assume there are m input attributes called X_1, X_2, … X_m
- Break dataset into n_Y smaller datasets called DS_1, DS_2, … DS_{ny}.
- Define DS_i = Records in which $Y=v_i$
- For each DS_i , learn Density Estimator M_i to model the input distribution among the $Y=v_i$ records.

How to build a Bayes Classifier

- Assume you want to predict output Y which has arity n_Y and values v_1, v_2, ... v_{ny}.
- Assume there are m input attributes called X_1, X_2, ... X_m
- Break dataset into n_Y smaller datasets called DS_1, DS_2, ... DS_{ny}.
- Define DS_i = Records in which $Y=v_i$
- For each DS_i , learn Density Estimator M_i to model the input distribution among the $Y=v_i$ records.
- M_i estimates P(X_1, X_2, ..., X_m | $Y=v_i$)

How to use a Bayes Classifier

- When a new set of input values ($X_1 = u_1$, $X_2 = u_2$, ... $X_m = u_m$) come along to be evaluated, predict the value of Y that makes P($Y=v_i$ | X_1, X_2, ... X_m) largest:

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\, P(Y = v \mid X_1 = u_1 \cdots X_m = u_m)$$

Bayes Classifier

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\, P(Y = v \mid X_1 = u_1 \wedge \cdots \wedge X_m = u_m)$$

$$P(Y = v \mid X_1 = u_1 \wedge \cdots \wedge X_m = u_m)$$

$$= \frac{P(X_1 = u_1 \wedge \cdots \wedge X_m = u_m \mid Y = v)P(Y = v)}{P(X_1 = u_1 \wedge \cdots \wedge X_m = u_m)}$$

$$= \frac{P(X_1 = u_1 \wedge \cdots \wedge X_m = u_m \mid Y = v)P(Y = v)}{\sum_{j=1}^{n_Y} P(X_1 = u_1 \wedge \cdots \wedge X_m = u_m \mid Y = v_j)P(Y = v_j)}$$

Bayes Classifiers in a nutshell

1. Learn the distribution over inputs for each value of Y.
2. This gives P(X_1, X_2, … X_m | $Y=v_i$).
3. Estimate P($Y=v_i$). as fraction of records with $Y=v_i$.
4. For a new prediction:

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\, P(Y = v \mid X_1 = u_1 \wedge \cdots \wedge X_m = u_m)$$
$$= \underset{v}{\operatorname{argmax}}\, P(X_1 = u_1 \wedge \cdots \wedge X_m = u_m \mid Y = v)P(Y = v)$$

How should we estimate these conditional densities?

Conditional Independence

- Let E_1, E_2, and E_3 be events. Then E_1 and E_2 are conditionally independent given E_3 if and only if

$$P(E_1 | E_2 \wedge E_3) = P(E_1 | E_3)$$

- Means that when E_3 is known to be true, knowing that E_2 is also true has no effect on the probability that E_1 is true.

Naïve Bayes Classifier

- General Bayes classifier:

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\, P(X_1 = u_1 \wedge \cdots \wedge X_m = u_m \mid Y = v) P(Y = v)$$

- Make the naïve assumption that the attributes are *mutually conditionally independent given the class*. This leads to the following drastic simplification:

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\, P(Y = v) \prod_{j=1}^{m} P(X_j = u_j \mid Y = v)$$

Naïve Bayes Classifier

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\, P(Y=v)\prod_{j=1}^{m} P(X_j = u_j \mid Y=v)$$

Technical Hint:
If you have 10,000 input attributes that product will underflow in floating point math. You should use logs:

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}}\left(\log P(Y=v) + \sum_{j=1}^{m} \log P(X_j = u_j \mid Y=v) \right)$$

PlayTennis Example

- Have joint P(O, T, H, W, PT), where
  Outlook values are {sunny, overcast, rain}
  Temperature values are {hot, mild, cool}
  Humidity values are {high, normal}
  Wind values are {weak, strong}
  PlayTennis values are {yes, no}
- Total of 72 probabilities involved (71 free parameters)

PlayTennis example: naïve Bayes

- Just need 4 pairwise conditional densities:
  - P(Outlook | PlayTennis) [4 free params.]
  - P(Temperature | PlayTennis) [4 free params.]
  - P(Humidity | PlayTennis) [2 free params.]
  - P(Wind | PlayTennis) [2 free params.]
- Plus the prior P(PlayTennis) [1 free param.]
- Total of only 13 free parameters (22 probability values) involved.

PlayTennis example: estimating the required conditional probabilities

For example:

$$P(O=s \mid PT = y) = \frac{\text{\# of data with } O=s \wedge PT=y}{\text{\# of data with } PT=y}$$

$$= 2/9$$

In all, need to determine 22 probability estimates.

More Facts About Bayes Classifiers

- Many other density estimators can be slotted in*.
- Density estimation can be performed with real-valued inputs*
- Bayes Classifiers can be built with real-valued inputs*
- Rather Technical Complaint: Bayes Classifiers don't try to be maximally discriminative---they merely try to honestly model what's going on*
- Zero probabilities are painful for Joint and Naïve. A hack (justifiable with the magic words "Dirichlet Prior") can help*.
- Naïve Bayes is wonderfully cheap. And survives 10,000 attributes cheerfully!

*See future Andrew Lectures

More Facts About Bayes Classifiers

Coming attraction: We'll learn all about Dirichlet priors and how they help us avoid zero probability estimates in the next set of slides.

- Zero probabilities are painful for Joint and Naïve. A hack (justifiable with the magic words "Dirichlet Prior") can help*.
- Naïve Bayes is wonderfully cheap. And survives 10,000 attributes cheerfully!

*See future Andrew Lectures

What you should know

- Probability
  - Fundamentals of Probability and Bayes Rule
  - What’s a Joint Distribution
  - How to do inference (i.e. $P(E_1|E_2)$) once you have a JD
- Bayesian Hypothesis Learning
  - MAP hypotheses

What you should know

- Density Estimation
  - What is DE and what is it good for
  - How to learn a Joint DE
  - How to learn a naïve DE
- Bayes Classifiers
  - How to build one
  - How to predict with a BC
  - Contrast between naïve and joint BCs