Pattern Recognition and Machine Learning

James L. Crowley

ENSIMAG 3 MMIS First Semester 2010/2011

Lesson 2 6 October 2010

Bayesian Probability

Contents

Source:
"Pattern Recognition and Machine Learning", C. M. Bishop, Springer Verlag, 2006.

Notation

x	a variable
X	a random variable (unpredictable value)
$\vec{x}$	A vector of D variables.
$\vec{X}$	A vector of D random variables.
D	The number of dimensions for the vector $\vec{x}$ or $\vec{X}$
E	An observation. An event.
k	Class index
K	Total number of classes
ω_k	The statement (assertion) that E $\in T_k$
M	Total number of examples.
$\{X_m\}$	A set of M examples of the feature X for events
$\{T_m\}$	A set of class labels (indicators) for the samples
$\{X_m^k\}$	A set of M_k examples for the class k. $\{X_m\} = \bigcup_{k=1,K} \{X_m^k\}$
M_k	Number of examples for the class k. (think M = Mass) $M = \sum_{k=1}^{K} M_k$

Probability and Uncertainty

One could even say that recognition is a problem of assigning signals to categories in the presence of uncertainty. The core problem of recognition is uncertainty.

We can distinguish two separate kinds of uncertainties: Confidence and Accuracy (Precision).

Confidence: Freedom from doubt, belief in the truth of a proposition.
Accuracy : Reproducibility of a measurement.

Confidence concerns the truth of a statement. The proposition is generally formalized as a predicate. It is either true or false.

Accuracy concerns a selecting an entity from an ordered set. Generally there is some order between the possible values with an associated distance metric. The accuracy refers to the size of a subset of possible values or the distance spanned by possible values.

In popular language, accuracy is often confused with precision.
In informatics:
Accuracy is the degree to which a measurement can be reproduced.
Precision is the detail with which a measurement is represented.

For example, a measurement may be represented with 32 bits of precision, but be accurate to only 8 bits (1 part in 256).
In common usage, precision and accuracy are often used for the same concept.

Probability is a powerful tool for both Confidence and Accuracy.

Both confidence and precision may be addressed in using Bayesian probabilities.

Probability as Frequency of Occurence.

A frequency based definition of probability is sufficient for many practical problems.

Suppose we have M observations of random events, $\{E_m\}$, for which M_k of these events belong to the class k. The probability that one of these observed events belongs to the class k is:

$$\Pr(E \in T_k) = \frac{M_k}{M}$$

If we make new observations under the same observations conditions (ergodicity), then it is reasonable to expect the fraction to be the same. However, because the observations are random, there may be differences. These differences will grow smaller as M grows larger.

The average (root-mean-square) error for

$$\Pr(E \in T_k) = \frac{M_k}{M}$$

will be proportional to M_k and inversely proportional to M.

Axiomatic Definition of probability

An axiomatic definition makes it possible to apply analytical techniques to the design of classification systems. Only three postulates (or axioms) are necessary:

In the following, let E be an event, let S be the set of all events, and let T_k be set of events that belong to class k with K total classes. $S = \bigcup_{k=1,K} T_k$

Postulate 1 : $\forall\, T_k \in S \;:\; p(E \in T_k) \geq 0$

Postulate 2 : $p(E \in S) = 1$

Postulate 3 :

$\forall\, T_i, T_j \in S$ such that $T_i \cap T_j = \varnothing$: $p(E \in T_i \cup T_j) = p(E \in T_i) + p(E \in T_j)$

A probability function is any function that respect these three axioms.

A probability is the truth value produced by a probability function.

Histogram Representation of Probability

We can use histograms both as a practical solution to many problems and to illustrate fundamental laws of axiomatic probability.

When we have K classes of events, we can build a table of frequency of occurrence for events from each class $h(E \in T_k)$.

The table of "frequency of occurrence" is also known as a "histogram", h(x).
The existence of computers with gigabytes of memory has made the computation of such tables practical.

The table h() can be implemented as a hash table, using the labels for each class as a key. Alternatively, we can map each class onto K natural numbers k <- T_k

$$\forall m=1, M \ : \text{if } E_m \in T_k \text{ then } h(k) := h(k) + 1;$$

After M events, given a new event, E,

$$p(E \in T_k) = p(k) = \frac{1}{M} h(k)$$

Problem: How many observations, M, do we need?

Answer: Given N possible values of X, h(x) has Q = N cells.

For M observations, in the worst case the RMS error between an estimated h(X) and the true h(x) is proportional to O(Q/M).

For most applications, $M \geq 10\,Q$ (10 samples per "cell") is reasonable.

Bayesian Probability

Bayesian probability can be seen as an extension of logic that enables reasoning with uncertain statements. Bayesian probability interprets probability as "a measure of a state of knowledge", rather than as "frequency of occurrence".

In Bayesian probability, the confidence of a proposition is represented by a probability number between 0 and 1.

To evaluate the confidence of a hypothesis, we determine a prior probability
This prior is then updated by observing new evidence.

The Bayesian interpretation provides a standard set of procedures and formulae to perform this calculation.

"Bayesian" refers to the 18th century mathematician and theologian Thomas Bayes (1702–1761), who provided the first mathematical treatment of a non-trivial problem of Bayesian inference. Bayesian probability was made popular by Simon Laplace in the early 19th century.

The rules of Bayesian logic can be justified by requirements of rationality and consistency and interpreted as an extension of logic. Many modern machine learning methods are based on objectivist Bayesian principles.

Although Bayesian logic is based on axiomatic probability, we can use histograms to illustrate the fundamental rules.

Illustrating Bayes Rule with Histograms

Suppose we have a set of events described by a pair of properties.
For example, consider the your grade in 2 classes C1 and C2.
Assume your grade is a letter grade from the set {A, B, C, D, F}.

We can build a 2 dimensional hash table, where each letter grade acts as a key into the table $h(x_1, x_2)$.

This hash table has $Q= 5 \times 5 = 25$ cells.

Each student is an observation with a pair of grades (x_1, x_2).

$\forall m=1, M$: if $h(x_1, x_2) := h(x_1, x_2) + 1$;

Question: How many students are needed to fill this table?
Answer $M \geq 10Q = 250$.

An example, consider the table as follows:

$X_2 \setminus X_1$	A	B	C	D	F	Total
A	2	5	3	1		11
B	5	16	8	1		30
C	2	12	20	3	1	38
D		2	6	2	2	12
F			4	4	1	9
Total	9	35	41	11	4	100

Any cell, (x_1, x_2) represents the probability that a student got grade X_1 for course C_1 and grade X_2 for course C_2.

$$p(X_1 = x_1 \wedge X_2 = x_2) = \frac{1}{M} h(x_1, x_2)$$

Let us note the sum of column i as c_i and sum of row j as r_j and the value of cell i,j as $h_{i,j}$

$$c_i = \sum_{j=A,B,\ldots F} h(i,j) \qquad r_j = \sum_{i=A,B,\ldots F} h(i,j) \qquad h_{ij} = h(i,j)$$

for example $r_B = 30$, $c_B = 35$, $h_{BB} = 16$

From this table we can easily see three fundamental laws of probability:

Sum Rule: $$p(X_1 = x_1) = \sum_{x_2=A,B,\ldots,F} p(X_1 = x_1, X_2 = x_2) = \frac{1}{M} \sum_{x_2=A,B,\ldots,F} h(x_1, x_2) = \frac{r_{x1}}{M}$$

example: $$p(x_1 = B) = \sum_{x_2=A,B,\ldots,F} p(x_1 = B, x_2) = \frac{1}{M} \sum_{x_2=A,B,\ldots,F} h(B, x_2) = \frac{r_B}{M} = \frac{30}{100}$$

from which we derive the sum rule:

$$p(X_1 = x_1) = \sum_{X_2} p(X_1 = x_1, X_2 = x_2)$$

or more simply

$$p(X_1) = \sum_{X_2} p(X_1, X_2)$$

This is sometimes called the "marginal" probability, obtained by "summing out" the other probabilities.

Conditional probability :
We can define a "conditional" probability as the fraction of one probability given another.

$$p(X_1 = i \mid X_2 = j) = \frac{h_{ij}}{r_j} \quad \text{and} \quad p(X_2 = j \mid X_1 = i) = \frac{h_{ij}}{c_i}$$

For example.

$$p(X_1 = B \mid X_2 = B) = \frac{h_{BB}}{r_B} = \frac{16}{30} \text{ and } p(X_2 = B \mid X_1 = B) = \frac{h_{BB}}{c_B} = \frac{16}{35}$$

From this, we can derive Bayes rule :

$$p(X_1 = i \mid X_2 = j) \cdot p(X_2 = j) = \frac{h_{ij}}{r_j} \cdot r_j = h_{ij} = \frac{h_{ij}}{c_i} \cdot c_i = p(X_2 = j \mid X_1 = i) \cdot p(X_1 = i)$$

or more simply

$$p(X_1 \mid X_2) \cdot p(X_2) = p(X_2 \mid X_1) \cdot p(X_1)$$

or more commonly written:

$$p(X_1 \mid X_2) = \frac{p(X_2 \mid X_1) \cdot p(X_1)}{p(X_2)}$$

Product Rule:

We can also use the histogram to derive the product rule.

Note that $p(X_1 = i, X_2 = j) = h_{ij}$

$$p(X_1 = i \mid X_2 = j) = \frac{h_{ij}}{r_i}$$

and

$$p(X_1, X_2) = p(X_1 \mid X_2) \cdot p(X_2)$$

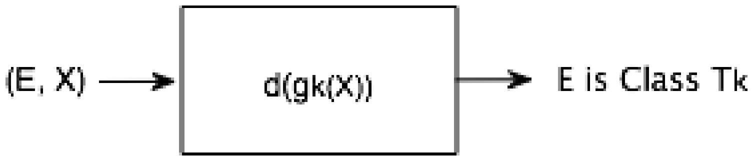
These rules show up frequently in machine learning and Bayesian estimation.

Co-occurrence of classes and features

We can also mix classes and features in a table of co-occurrences.

Features are properties of events that be used to classify the event.

Our problem is to build a box that maps a set of features $\vec{X}$ into a class T_k from a set of K possible Classes.



Just as we can compute the table of probabilities for classes of events, we can compute probabilities for values of feature, whether the values are symbolic or numeric.

This is equivalent to considering each value of the feature as a class. We can then compute a frequency table of possible values.

For example, suppose we have events E described by a feature X that can take on one of N values.

Given a training set of M events $\{X_m\}$ where each event is labeled with a ground truth class label $\{T_m\} \in \{1, 2, \ldots, K\}$.

We can build a NxK frequency table h(x, k): $\forall m=1, M$: $h(T_m, X_m) = h(T_m, X_m)+1$

The table h(k,x) gives the joint probability $p(T_m=k, X=x)$

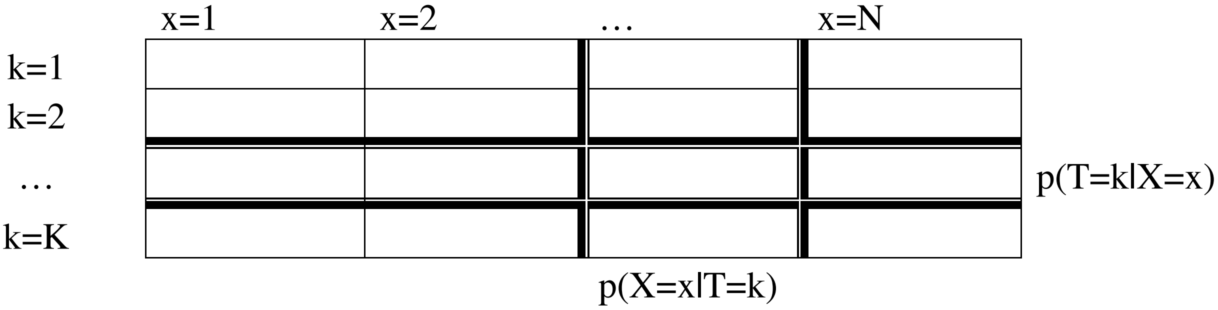
As before:

$$P(X = x) = \frac{1}{M}\sum_{k=1}^{K} h(k,x)$$

$$P(T = k) = \frac{1}{M}\sum_{x=1}^{N} h(k,x)$$

$$P(X = x \mid T = k) = \frac{h(k,x)}{\sum_{k=1}^{K} h(k,x)}$$

$$P(T = k \mid X = x) = \frac{h(k,x)}{\sum_{x=1}^{N} h(k,x)}$$



Note that we did not need to use numerical values for T or X.

If the features are symbolic, h(T,X) is a hash, and the feature and class labels act as a hash key. In this case h(T,X) is called a bag.

"Bag of Features" methods are increasingly used for learning and recognition.

Probability of Numerical Features.

Frequency tables can, of course, also be used for numerical features.
For numerical features, there is a natural order relation (e.g. ">") between the values of the features. This order relation makes possible additional operations.

For example, suppose we have M observations of an event described by a feature, X, where X can have one of N values from set $X \in [x_{min}, x_{max}]$.
To simplify, we can map X onto the natural numbers {1, 2, …, N}

Is observed features are continous values, we can always map these to the natural numbers:

if $X \leq x_{min}$ then n = 1
if $X \geq x_{max}$ then n = N.
else $n = Trunc(N \cdot \frac{X - x_{\min}}{x_{\max} - x_{\min}}) + 1$

We can use a frequency tables, h(x) with Q=N cells to compute the probability of obtaining a particular value.

Given M observations, and build a table of frequencies for each value.

$\forall m=1, M$: if $X_m \in x$ then h(x) := h(x) + 1;

after M events, the probability of an observation having value X = x is:

$$p(X = x) = \frac{1}{M} h(x)$$

Consider a problem of assigning an event E to one of two classes, {A, B} based on a numerical "feature" X, and that the numerical feature X is mapped to the natural numbers [1, ..., N].

Assume that we have a Training Sets with M_A observations of class A: $\{X_m^A\}$
and M_B observations of class B $\{X_m^B\}$. The total training data is composed of

$M=M_A+M_B$ observations $\{X_m\} = \{X_m^A\} \cup \{X_m^B\}$

We can build frequency tables: $h_A(x)$, $h_B(x)$and h(x).

```
∀m=1, M_A : if $X_m^A$ = x then $h_A(x) := h_A(x) + 1$;
∀m=1, M_b : if $X_m^B$ = x then $h_B(x) := h_B(x) + 1$;
∀m=1, M : if $X_m$ = x then $h(x) := h(x) + 1$;
```

Note that $h(x) = h_A(x) + h_B(x)$.

then

$$p(X=x) = \frac{1}{M} h(x)$$

$$p(X=x \mid E=A) = \frac{1}{M_A} h_A(x)$$

$$p(X=x \mid E=B) = \frac{1}{M_B} h_B(x)$$

and $$p(E=A) = \frac{M_A}{M} = \frac{M_A}{M_A + M_B}$$

From Bayes Rule:

$$p(E=A \mid X=x) = \frac{p(X=x \mid E=A)p(E=A)}{p(X=x)} = \frac{\frac{1}{M_A} h_A(x) \frac{M_A}{M}}{\frac{1}{M} h\ (x)} = \frac{h_A(x)}{h\ (x)}$$

The probability that event E is class A given feature X is x is simply the ratio of the histogram for class A divided by the histogram of X for all classes.

Thus we can extend Bayes Rule to computing the probability of a proposition based on numerical values of features.

Histograms and the Curse of Dimensionality

Computers and the Internet make it possible to directly apply histograms to very large amounts of data, and to consider very large feature sets. For such applications it is necessary to master the size of the histogram and the quantity of data.

Assume a feature vector $\vec{X}$, composed of D features, where each feature has one of N possible values.

The histogram "capacity" is the number of cells $Q=N^{D}$. Obviously, this grows exponentially with D. It is often convenient to reason in powers of 2 here.

Note 2^{10}=Kilo, 2^{20}=Meg, 2^{30}=Giga, 2^{40}=Tera, 2^{50}=Peta,

Here is a table of numbers of cells, Q, in a histogram of D dimensions of N values.

N \ d	1	2	3	4	5	6
2	2^{1}	2^{2}	2^{3}	2^{4}	2^{5}	2^{6}
4	2^{2}	2^{4}	2^{6}	2^{8}	2^{10} =1 Kilo	2^{12} =2 Kilo
8	2^{3}	2^{6}	2^{9}	2^{12}	2^{15}	2^{18}
16	2^{4}	2^{8}	2^{12}	2^{16}	2^{20} = 1 Meg	2^{24} = 4 Meg
32	2^{5}	2^{10} =1 Kilo	2^{15}	2^{20} = 1 Meg	2^{25}	2^{30} = 1 Gig
64	2^{6}	2^{12}	2^{18}	2^{24}	2^{30} = 1 Gig	2^{36}
128	2^{7}	2^{14}	2^{21} = 2 Meg	2^{28}	2^{35}	2^{42} =2 Tera
256	2^{8}	2^{16}	2^{24}	2^{32} = 2 Gig	2^{40} = 1 Tera	2^{48}

In this case, the RMS error between a histogram and the underlying density is

$$E_{RMS} (h(X)-P(X)) = O(Q/M).$$

As a rule, it is recommended to have 10 samples per cell. $M \geq 10 Q$.
The worst case occurs when the true underlying density is uniform.

For example, for D=5 features each with N = 32 values, the histogram has 1 Meg cells and you need 10 Meg of data.

For D= 6 features with N=64 values, h() has 1 Gig of cells and you need 10 Giga of samples.
For higher numbers of values or features, it is more convenient to work with probability densities.