Intelligent Systems: Reasoning and Recognition

James L. Crowley

ENSIMAG 2 / MoSIG M1 Second Semester 2013/2014

Lesson 14 4 April 2014

Non-parametric Methods for Classification

Notation 2

Bayesian Classification.....3

Histograms as a representation for probability 4

Supervised Learning 4
Multi-dimensional Histograms 4
Capacity of Multi-dimensional Histograms 5
Variable size histogram cells 6

Kernel Density Estimators.....7

K Nearest Neighbors 9

Probability Density Functions 10

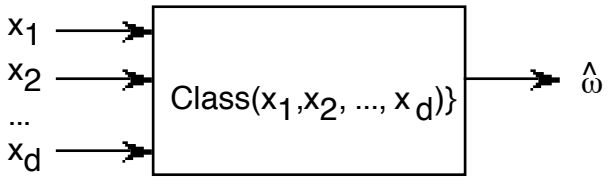
Bibliographical sources:
"Pattern Recognition and Machine Learning", C. M. Bishop, Springer Verlag, 2006.
"Pattern Recognition and Scene Analysis", R. E. Duda and P. E. Hart, Wiley, 1973.

Notation

x	a variable
X	a random variable (unpredictable value)
N	The number of possible values for X (Can be infinite).
$\vec{x}$	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.
C_k	The class k
k	Class index
K	Total number of classes
ω_k	The statement (assertion) that $E \in C_k$
$p(\omega_k) = p(E \in C_k)$	Probability that the observation E is a member of the class k. Note that $p(\omega_k)$ is lower case.
M_k	Number of examples for the class k. (think M = Mass)
M	Total number of examples. $M = \sum_{k=1}^{K} M_k$
$\{X_m^k\}$	A set of M_k examples for the class k. $\{X_m\} = \bigcup_{k=1,K} \{X_m^k\}$
$p(X)$	Probability density function for X
$p(\vec{X})$	Probability density function for $\vec{X}$
$p(\vec{X} \mid \omega_k)$	Probability density for $\vec{X}$ the class k. $\omega_k = E \in T_k$.
Q	Number of cells in h(*n*). $Q = N^D$
P	A sum of V adjacent histogram cells: $P = \sum_{\vec{X} \in V} h(\vec{X})$

Bayesian Classification

Our problem is to build a box that maps a set of features $\vec{X}$ from an Observation, E into a class C_k from a set of K possible classes.



Let ω_k be the proposition that the event belongs to class k: $\omega_k = E \in C_k$

ω_k Proposition that event E $\in$ the class k

In order to minimize the number of mistakes, we will maximize the probability that $\omega_k \equiv E \in C_k$

$$\hat{\omega}_k = \arg\max_k \left\{ \Pr(\omega_k \mid \vec{X}) \right\}$$

Our primary tool for this is Baye's Rule : $P(\omega_k \mid \vec{X}) = \dfrac{P(\vec{X} \mid \omega_k) P(\omega_k)}{P(\vec{X})}$

To apply Baye's rule, we require a representation for the probabilities $P(\vec{X} \mid \omega_k)$, $P(\vec{X})$, and $p(\omega_k)$.

The term $p(\omega_k)$ is a number that represents the a-priori probability of encountering an event of class K. For a training set of M samples of which M_k are from class k, this is simply the frequency of occurrence of class k.

$$P(\omega_k) = \frac{M_k}{M}$$

The terms $P(\vec{X} \mid \omega_k)$, $P(\vec{X})$ are more subtle.

We have already seen how to use histograms to represent $P(\vec{X} \mid \omega_k)$ and $P(\vec{X})$
Today will look at two non-parametric representations for $P(\vec{X} \mid \omega_k)$ and $P(\vec{X})$

1) Kernel Density Estimators
2) K-Nearest Neighbors

Histograms as a representation for probability

Supervised Learning

We will use a set of labeled "training set" of samples to estimate the probabilities $p(\vec{X})$, $p(\vec{X}\,|\,\omega_k)$, and $P(\omega_k)$. This is referred to as "supervised learning".

Assume that we have *K* classes.
For each class we have a set of M_k sample events $S_k = \{\vec{x}_m^k\}$.

The union of the training samples for each class gives us our training set:

$S = \{\vec{x}_m\} = \bigcup_{k=1,K} \{\vec{x}_m^k\}$ composed of $M = \sum_{k=1}^{K} M_k$ samples (think M = Mass)

In the simplest cases, we can use histogram (tables of frequencies) to represent the probabilities. Alternatively, we can present $p(\vec{X})$, $p(\vec{X}\,|\,\omega_k)$ as Probability Density Functions.

Multi-dimensional Histograms

Recall: ∀m=1 to M: $h(X_m) \leftarrow h(X_m)+1$

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

As a representation of probability, histograms have advantages and disadvantages.

Advantages include:
1) Because the $M = \sum_{\vec{X}} h(\vec{X})$ we are sure that $\sum_{\vec{X}} p(\vec{X}) = 1$
2) Histograms have a fixed size, Q, independent of the quantity of data.
It is not necessary to store the sample data, only the histogram.
3) Histograms can be composed and used incrementally.

The disadvantages are that
1) Each feature must be quantized over a limited range of N values.
(or from a predefined set of N symbols).
2) We need M >> Q = N^D data samples.
3) There are discontinuities at the boundaries of each cell.

Capacity of Multi-dimensional Histograms

Computers and the Internet make it possible to directly apply histograms to very large sets 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 the number 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}

For example, for D=4 features each with N = 32=2^5 values, the histogram has $2^{4x5} = 2^{20}$ = 1 Meg cells and you need 8 Meg = 2^{23} samples of data.

For D= 5 features with N=64=2^6 values, h() has $2^{5x6} = 2^{30}$ = 1 Gig of cells and you need 2^{33} = 8 Giga of samples.

For higher numbers of values or features, it is more convenient to work with probability densities.

Variable size histogram cells

Suppose that we have a D-dimensional feature vector $\vec{X}$ with each feature quantized to N possible values, and suppose that we represent $P(\vec{X})$ as a D-dimensional histogram h($\vec{X}$). Let us fill the histogram with M training samples $\{\vec{X}_m\}$.

Let us define the volume of each cell as 1.
The volume for any block of V cells is V.
Then the volume of the entire space is $Q=N^D$.

If the quantity of training data is too small, ie $M < 8Q$ we can combine adjacent cells so as to amass enough data for a reasonable estimate.

Suppose we merge V adjacent cells such that we obtain a combined sum of P.

$$P = \sum_{\vec{X} \in V} h(\vec{X})$$

The volume of the combined cells would be V
The probability $p(\vec{X})$ for $\vec{X} \in V$ is

$$p(\vec{X} \in V) = \frac{P}{MV}$$

This is typically written as: $p(\vec{X}) = \frac{P}{MV}$

We can use this equation to develop two alternative non-parametric methods.

Fix V and determine P => Kernel density estimator.
Fix P and determine V => K nearest neighbors.

(note in most developments the symbol “K” is used for the sum the cells. This conflicts with the use of K for the number of classes. Thus we substitute the symbol P for the sum of adjacent cells).

Kernel Density Estimators

For a Kernel density estimator, we will represent each data point with a kernel function $k(\vec{X})$.

Popular Kernel functions are

a hypercube centered of side w
a sphere of radius w
a Gaussian of standard deviation w.

We can define the function for the hypercube as

$$k(\vec{u}) = \begin{cases} 1 & if \ |u_d| \le 1/2 \ \text{ for all } d = 1,...,D \\ 0 & otherwise \end{cases}$$

This is called a Parzen window.

For a position $\vec{X}$, the total number of points lying with a cube with side w will be:

$$P = \sum_{m=1}^{M} k\left(\frac{\vec{X} - \vec{X}_m}{w}\right)$$

The volume of the cube $V = \frac{1}{w^D}$.

Thus the probability $p(\vec{X}) = \frac{P}{MV} = \frac{1}{Mw^D} \sum_{m=1}^{M} k\left(\frac{\vec{X} - \vec{X}_m}{w}\right)$

The Hypercube has a discontinuity at the boundaries. We can soften this using a triangular function evaluated on a sphere.

$$k(\vec{u}) = \begin{cases} 1 - 2\|\vec{u}\| & if \ \|\vec{u}\| \le 1/2 \ \text{ for all } d = 1,...,D \\ 0 & otherwise \end{cases}$$

Even better is to use a Gaussian kernel with standard deviation $\sigma = w$.

$$k(\vec{u}) = e^{-\frac{1}{2}\frac{\|\vec{u}\|^2}{w^2}}$$

We can note that the volume is $V = (2\pi)^{D/2} w^D$

In this case $p(\vec{X}) = \frac{P}{MV} = \frac{1}{M(2\pi)^{D/2} w^D} \sum_{m=1}^{M} k\left(\vec{X} - \vec{X}_m\right)$

This corresponds to placing a Gaussian over each point and summing the Gaussians.

In fact, we can choose any function $k(\vec{u})$ as kernel, provided that

$k(\vec{u}) \geq 0$ and $\int k(\vec{u}) d\vec{u} = 1$

The Gaussian Kernel tends to be popular for Machine Learning.

K Nearest Neighbors

For K nearest neighbors, we hold P constant and vary V. (We have used the symbol P for the number of neighbors, rather than K to avoid confusion with the number of classes).

As each data samples, $\vec{X}_m$, arrives, we construct a tree structure (such as a KD Tree) that allows us to easily find the P nearest neighbors for any point .

To compute $p(\vec{X})$ we need the volume of a sphere of radius $\|\vec{X} - \vec{X}_K\|$ in D dimensions. This is:

$$V = C_D \left\| \vec{X} - \vec{X}_K \right\|^D \qquad \text{where} \qquad C_D = + \frac{\pi^{\frac{D}{2}}}{\Gamma\left(\frac{D}{2}+1\right)}$$

Where $\Gamma(n) = (n-1)!$

For odd D, use a table to determine $\Gamma\left(\frac{D}{2}+1\right)$

Then as before: $p(\vec{X}) = \frac{P}{MV}$

Probability Density Functions

The alternative to a non-parametric representation is to use a function to represent $P(\vec{X} \mid \omega_k)$ and $P(\vec{X})$. Such a function is referred to as a "Probability Density Function" or PDF.

A probability density function of a continuous random variable is a function that describes the relative likelihood for this random variable to occur at a given point in the observation space. The integral of a pdf gives a probability.

Definition: "Likelihood" is a relative measure of belief or certainty.
Note: Likelihood is not probability. We will use the "likelihood" to determine the parameters for parametric models of probability density functions. To do this, we first need to define probability density functions.

A probability density function, $p(\vec{X})$, is a function of a continuous variable or vector, $\vec{X} \in R^D$, of random variables such that :

1) $\vec{X}$ is a vector of D real valued random variables with values between $[-\infty, \infty]$
2) $\int_{-\infty}^{\infty} p(\vec{X}) = 1$

Note that, $p(\vec{X})$ is NOT a number but a continuous function. To obtain a probability we must integrate over some volume V of the D dimensional feature space.

$$P(\vec{X} \in V) = \int_V p(\vec{X}) d\vec{X}$$

This integral gives a number that can be used as a probability.

In the case of D=1, the probability that X is within the interval [A, B] is

$$p(X \in [A,B]) = \int_A^B p(x) dx$$

Consider $$p(\omega_k \mid \vec{X}) = \frac{p(\vec{X} \mid \omega_k)}{p(\vec{X})} p(\omega_k)$$

While $p(\vec{X})$ and $p(\vec{X} \mid \omega_k)$ are NOT numbers, their ratio IS a number.

The ratio of two pdfs can give a probability value!