If we take one dimensional case:

Samples are given

We need to find density function f(x) and we do not know form of f

We are going to apply simple idea to learn a piece wise constant approximation to f.

First we cut x axis into small intervals and build a function that is constant in each of the intervals

If f(x) = K over an interval [a, b] then

P[a<= X <= b] = K(b-a) The probability above is well approximated by the function of data points which fall in that interval.

Thus we can approximate f by the histogram of data.

Let R be region (sphere of some small radius around x ) let

If p is nearly constant over R then where

where V is the volume of R thus

Suppose out of n iid samples k samples fall in R then k is binomial with parameter n and . Since for large n binomial distribution sharply peaks around its mean. We expect the ratio k/n will be a very good estimate for the probability and hence for the smoothed density function. This estimate is especially accurate when n is sufficiently large.

or combining the two we could get

This is the basic idea of finding an approximation of f. At any x we take a small volume V around x and count the number of data samples that fall in this region. This gives approximate value of p(x) shown in above.

Choice of V affects the quality of approximation. For the approximation to be good we need V to be small.

But if V is very small, unless n is very large k may be zero most of the time. So choice of size of V is a compromise between these two requirements.

Let denote the volume when we have n examples and let and denote the corresponding values. Then for as we must have

, ,

We need to get the correct estimates

If then and hence Finally

If is to converge to p(x) three conditions are required to be met

1)

2)

3)

In practice we have only finite data. We choose size of V based on n.

Actually we have a choice of two approaches :

1) We can fix a V and calculate K, which is known as Parzen Window or Kernel Density Estimate

2) We can fix K and Calculate V, which is known as k-nearest neighbor method.