Introduction to local (nonparametric) density estimation methods

A slecture by Yu Liu for ECE 662 Spring 2014

1. Introduction

This slecture introduces two local density estimation methods which are Parzen density estimation and k-nearest neighbor density estimation. Local density estimation is also referred to as non-parametric density estimation. To make things clear, let's first look at parametric density estimation. In parametric density estimation, we can assume that there exists a density function which can be determined by a set of parameters. The set of parameters are estimated from the sample data and are later used in designing the classifier. However, in some practical situations the assumption that there exists a parametric form of the density function does not hold true. For example, it is very hard to fit a multimodal probability distribution with a simple function. In this case, we need to estimate the density function in the nonparametric way, which means that the density function is estimated locally based on a small set of neighboring samples. Because of this locality, local (nonparametric) density estimation is less accurate than parametric density estimation. In the following text the word "local" is preferred over "nonparametric."

It is noteworthy that it is very difficult to obtain an accurate local density estimation, especially when the dimension of the feature space is high. So why do we bother using local density estimation? This is because our goal is not to get an accurate estimation, but rather to use the estimation to design a well performed classifier. The inaccuracy of local density estimation does not necessarily lead to a poor decision rule.

2. General Principle

In local density estimation the density function $p_n(x)$ can be approximated by

$$p_n(x) = \frac{k_n}{nv_n} \tag{1}$$

where v_n is the volume of a small region R around point x, n is the total number of samples x_i (i =1, 2…, n) drawn according to $p_n(x)$, and k_n is the number of x_i's which fall into region R. The reason why $p_n(x)$ can be calculated in this way is that $p_n(x)$ does not vary much within a relatively small region, thus the probability mass of region R can be approximated by $p_n(x)v_n$, which equals k_n/n.

Some examples of region R in different dimensions: i) line segment in one-dimension, ii) circle or rectangle in two-dimension, iii) sphere or cube in three-dimension, iv) hyper sphere or hypercube in d-dimension (d > 3).

Three conditions we need to pay attention to when using formula (1) are:

i) $\lim_{n\to\infty} v_n = 0$. This is because if v_n is fixed, then $p_n(x)$ only represents the average probability density as n grows larger, but what we need is the point probability density,

so we should have $v_n \to 0$ when $n \to \infty$.

ii) $\lim_{n\to\infty} k_n = \infty$. This is to make sure that we do not get zero probability density.

iii) $\lim_{n\to\infty} k_n / n = 0$. This is to make sure that $p_n(x)$ does not diverge.

3. Parzen Density Estimation

In Parzen density estimation v_n is directly determined by n while k_n is a random variable which denotes the number of samples that fall into v_n. Assume that the region R is a d-dimensional hypercube with its edge length h_n, thus

$$v_n = (h_n)^d$$

The equivalent conditions which meet the aforementioned three conditions are:

$$\lim_{n\to\infty} v_n = 0 \quad \text{and} \quad \lim_{n\to\infty} n v_n = \infty$$

Therefore v_n can be chosen as $v_n = h/\sqrt{n}$ or $v_n = h/\ln n$, where h is an adjustable constant. Now that the relationship between v_n and n is defined, the next step is to determine k_n. To determine k_n, we define a window function as follows:

$$\varphi\left(\frac{x - x_i}{h_n}\right) = \begin{cases} 1 & \dfrac{|x - x_i|}{h_n} < \dfrac{1}{2} \\ 0 & \dfrac{|x - x_i|}{h_n} > \dfrac{1}{2} \end{cases}$$

where x_i's (i = 1, 2, …, n) are the given samples and x is the point where the density is to be estimated. Thus we have

$$k_n = \sum_{i=1}^{n} \varphi\left(\frac{x - x_i}{h_n}\right)$$

$$p_n(x) = \frac{k_n/n}{v_n} = \frac{1}{n v_n} \sum_{i=1}^{n} \varphi\left(\frac{x - x_i}{h_n}\right)$$

The function φ is called a Parzen window function, which enables us to count the number of sample points in the hypercube with its edge length h_n.

According to [2], using hypercube as the window function may lead to discontinuity in the estimation. This is due to the superimposition of sharp pulses centered at the given sample points when h is small. To overcome this shortcoming, we can consider a more general form of window function rather than the hypercube. Note that if the following two conditions are met, the estimated $p_n(x)$ is guaranteed to be proper.

$$\varphi(x) \geq 0 \quad \text{and} \quad \int \varphi(x)dx = 1$$

Therefore a better choice of window function which removes discontinuity can be Gaussian window:

$$\varphi\left(\frac{x-x_i}{h_n}\right) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-x_i}{h_n}\right)^2\right)$$

The estimated density is given by

$$p_n(x) = \frac{1}{nv_n} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-x_i}{h_n}\right)^2\right) \tag{2}$$

Consider a one-dimension case, assume that $v_n = h/\sqrt{n}$, thus $h_n = v_n = h/\sqrt{n}$, where *h* is an adjustable constant. Substitute into formula (2) we have

$$p_n(x) = \frac{1}{nv_n} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-x_i}{h_n}\right)^2\right) = \frac{1}{h\sqrt{n}} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-x_i}{h/\sqrt{n}}\right)^2\right)$$

We can see that if *n* equals one, $p_n(x)$ is just the window function. If *n* approaches infinity, $p_n(x)$ can converge to any complex form. If *n* is relatively small, $p_n(x)$ is very sensitive to the value of *h*. In general small *h* leads to the noise error while large *h* leads to the over-smoothing error, which can be illustrated by the following example.

In this experiment samples are 5000 points on 2-D plane with Gaussian distribution. The mean vector is [1 2], and the covariance matrix is [1 0; 0 1]. Choose rectangle Parzen window with $h_n = 4/\sqrt[4]{n}$, thus $v_n = (h_n)^2 = 16/\sqrt{n}$. Fig. 1 shows the sample distribution. Fig. 2 shows the ideal probability density distribution. Fig. 3 shows the result of Parzen density estimation.

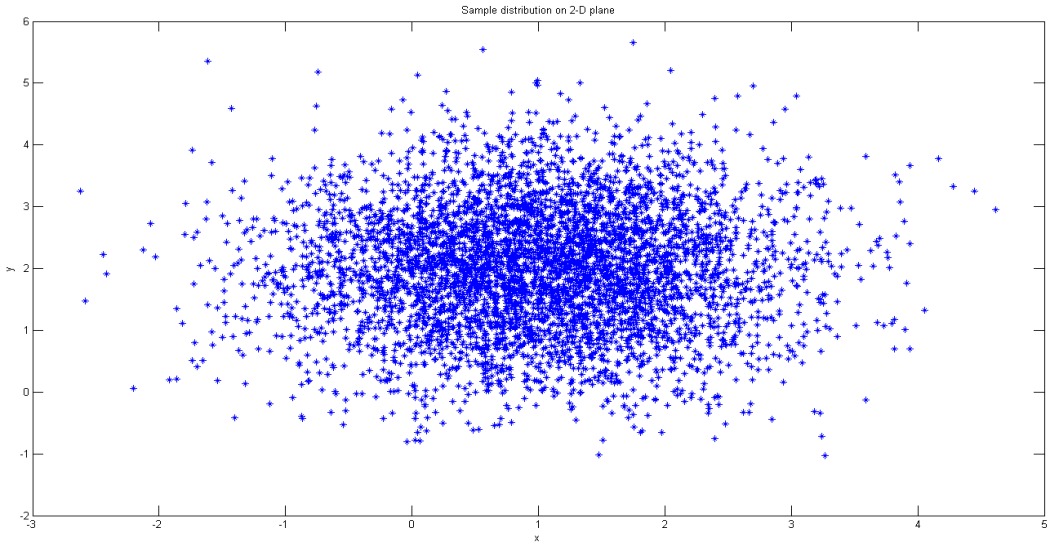

Figure 1. 5000 sample points on 2-D plane with Gaussian distribution

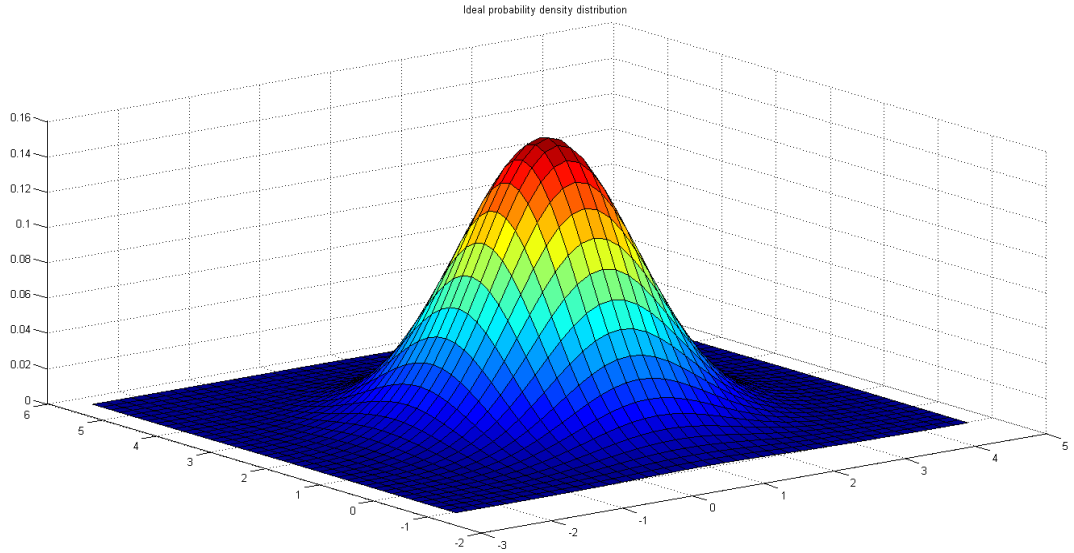

Figure 2. The ideal probability density distribution

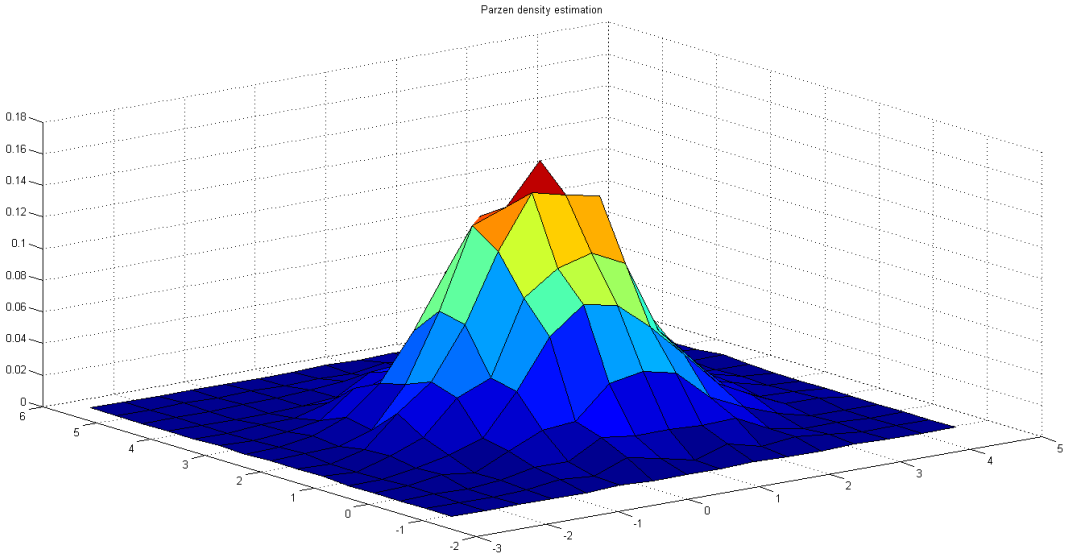

Figure 3. The result of Parzen density estimation

Next we change the value of h_n and see how it affects the estimation. Fig. 4 shows the result of Parzen density estimation when h_n is twice its initial value. Fig. 5 shows the result of Parzen density estimation when h_n is its initial value divided by two. We can see that the results agree with the aforesaid property of h_n.

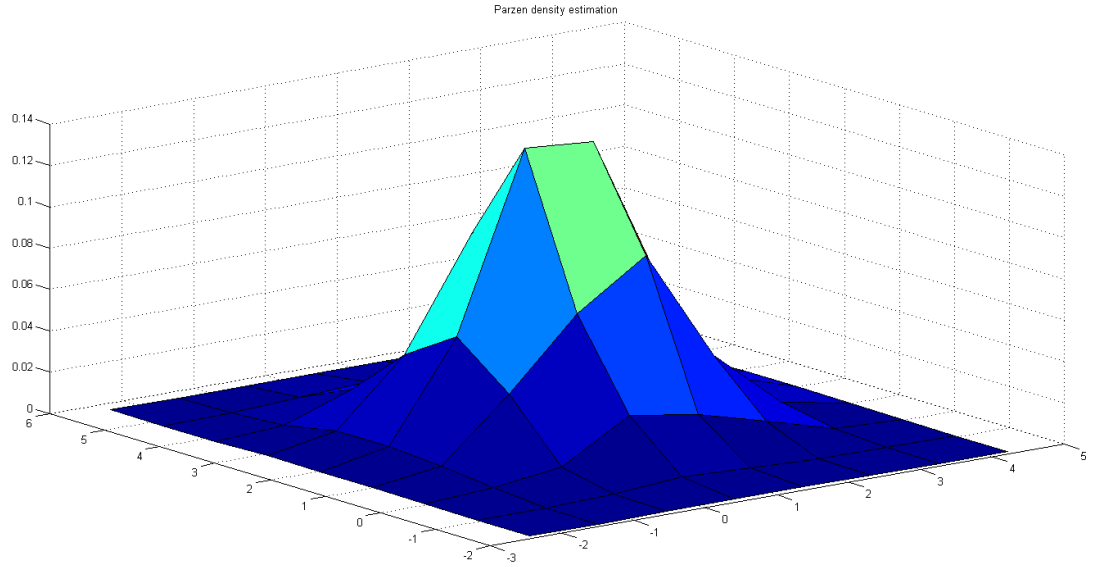

Figure 4. The result of Parzen density estimation when h_n is twice its initial value

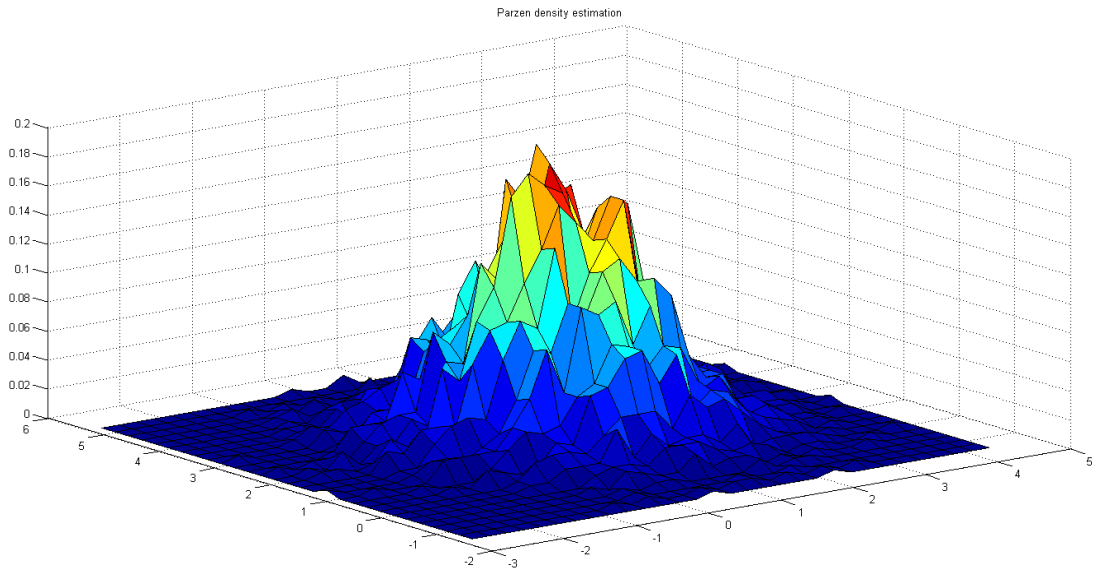

Figure 5. The result of Parzen density estimation when h_n is its initial value divided by two

To design a classifier using Parzen window method [3], we estimate the densities for each class and classify the test point by the label corresponding to the maximum posterior.

Below lists some advantages and disadvantages of Parzen density estimation:

Advantages: i) $p_n(x)$ can converge to any complex form when n approaches infinity; ii) applicable to data with any distribution.

Disadvantages: i) need a large number of samples to obtain an accurate estimation; ii) computationally expensive, not suitable for feature space with very high dimensions;

iii) the adjustable constant h has a relatively heavy influence on the decision boundaries when n is small, and is not easy to choose in practice.

4. K-Nearest Neighbor Density Estimation

In k-nearest neighbor density estimation (use acronym "k-NN" in the following text) k is directly determined by n while v is a random variable which denotes the volume that encompasses just k sample points inside v and on its boundary. If v is a sphere, it can be given by

$$v_k(x) = \min_h \left(\frac{\pi^{\frac{n}{2}} h^n}{\Gamma(\frac{n}{2}+1)} \right) = \frac{\pi^{\frac{n}{2}} (h_k)^n}{\Gamma(\frac{n}{2}+1)}$$

where h is the radius of the sphere with center x. h_k equals $\|x_{lk} - x\|$ where x_{lk} is the k^{th} closest sample point to x. Then the probability density at x is approximated by

$$\overline{p}(x) = \frac{k_1}{n v_k(x)} \tag{3}$$

where k_1 is number of sample points on the boundary of $v_k(x)$. Most of the time formula (3) can be rewritten as

$$\overline{p}(x) = \frac{k-1}{n v_k(x)} (k \geq 2)$$

It can be proved that $E[\overline{p}(x)] = p(x)$.

In Parzen density estimation v_n only depends on n and is the same for all the test points, while in k-NN v_n is smaller at high density area and is larger at low density area. This strategy seems more reasonable than the strategy to determine v_n in Parzen density estimation since now v_n is adaptive to the local density.

In practice, when we want to classify data using k-NN estimation, it turns out that we can get the posterior $p(w_i|x)$ directly without worrying about $p(x)$. If we have k samples fall into volume v around point x, and among the k samples there are k_i samples belonging to class w_i, then we have

$$p(w_i, x) = \frac{k_i / n}{v}$$

The posterior $p(w_i|x)$ is given by

$$p(w_i \mid x) = \frac{p(w_i, x)}{P(x)} = \frac{p(w_i, x)}{\sum_{j=1}^{m} p(w_j, x)} = \frac{k_i}{k} \tag{4}$$

where m is the number of classes. Formula (4) tells us one simple decision rule: the

class of a test point x is the same as the most frequent one among the nearest k points of x. Simple and intuitive, isn't it? Having said that, choosing k in k-NN is still a nontrivial problem as choosing h in Parzen density estimation. Small k leads to noisy decision boundaries while large k leads to over-smoothed boundaries, which is illustrated by the following example.

In this experiment samples are 400 randomly labeled (red or blue) points. The task is to find the classification boundaries under different k values. Fig. 6-9 show the results.

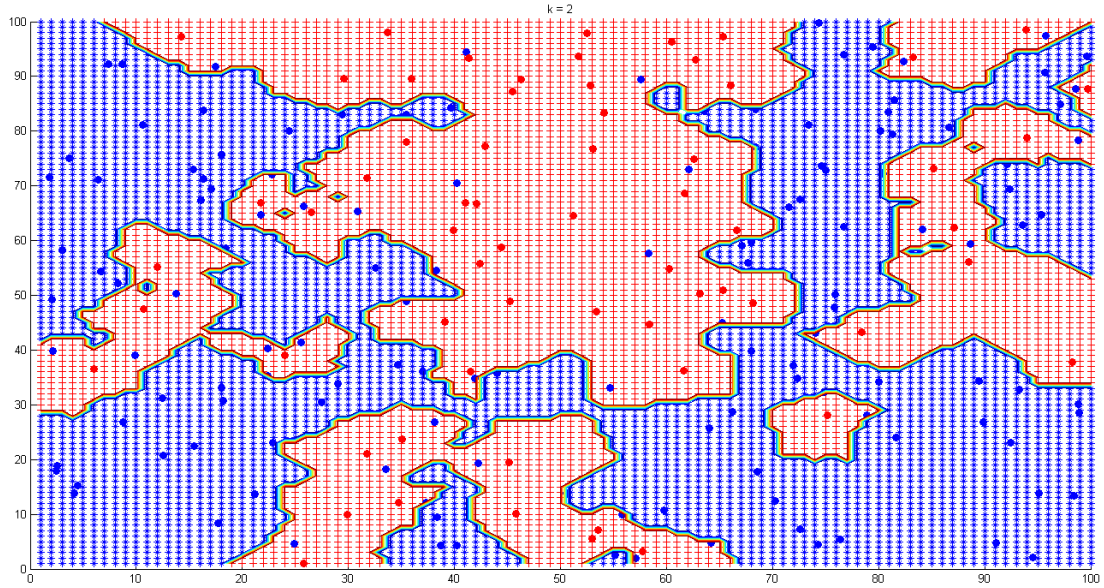

Figure 6. k-NN decision boundaries experiment (k=2)

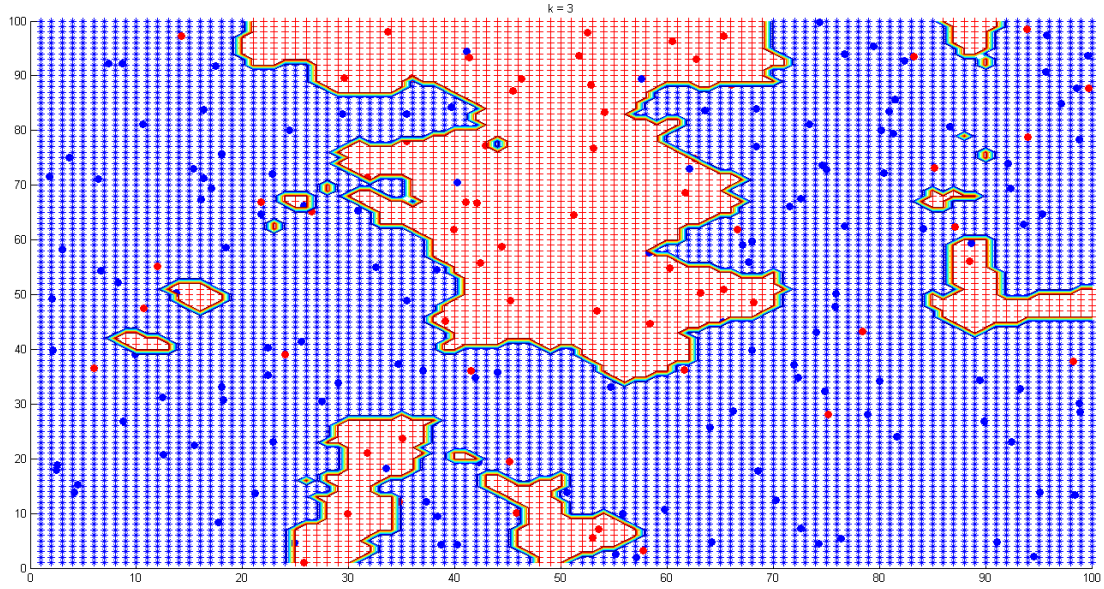

Figure 7. k-NN decision boundaries experiment (k=3)

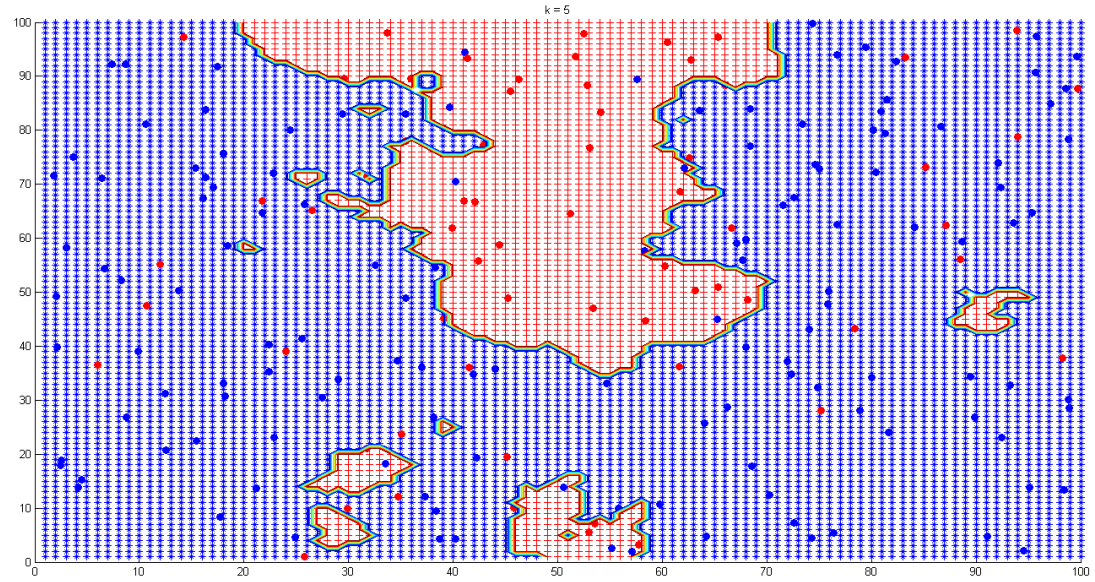

Figure 8. *k*-NN decision boundaries experiment (*k*=5)

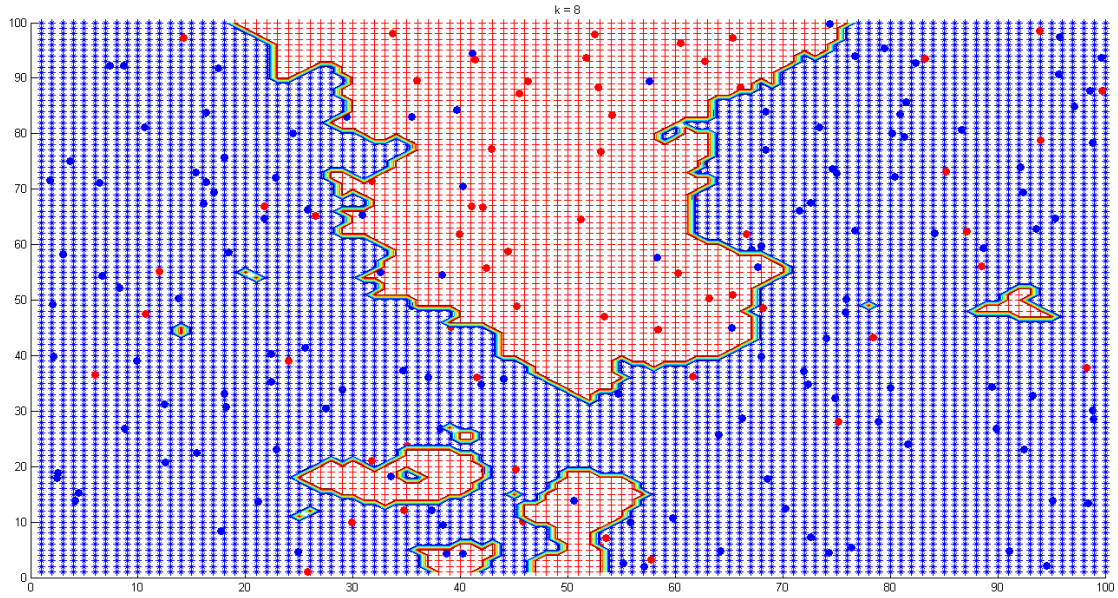

Figure 9. *k*-NN decision boundaries experiment (*k*=8)

In practice we can use cross-validation to choose the "best" *k*. Below lists some advantages and disadvantages of *k*-NN:

Advantages: i) decision performance is good if *n* is large enough; ii) applicable to data with any distribution; iii) simple and intuitive.

Disadvantages: i) need a large number of samples to obtain an accurate estimation, which is inevitable in local density estimation; ii) computationally expensive, low efficiency for feature space with very high dimensions; iii) choosing the "best" *k* is nontrivial.

5. Reference

[1] Mireille Boutin, “ECE662: Statistical Pattern Recognition and Decision Making Processes,” Purdue University, Spring 2014

[2] http://www.cse.buffalo.edu/~jcorso/t/CSE555/files/annote_28feb_nonprm.pdf

[3] http://www.csd.uwo.ca/~olga/Courses/CS434a_541a/Lecture6.pdf