北京航空航天大學
BEIHANG UNIVERSITY

# Synthetical Design of Bayesian Classifier

# Experiment Report

| | |
|---|---|
| School | School of Automatic Science |
| Major | Pattern Recognition and Intelligent System |
| Student | Tianxiang Lan |
| Student ID | 15231087 |
| Teacher | Yang Li |

April 22,2018

# 1 Introduction

Linear perceptron allow us to learn a decision boundary that would separate two classes. They are very effective when there are only two classes, and they are well separated. Such classifiers are referred to as discriminative classifiers. In contrast, generative classifiers consider each sample as a random feature vector, and explicitly model each class by their distribution or density functions. To carry out the classification, the likelihood function should be computed for a given sample which belongs to one of candidate classes so as to assign the sample to the class that

is most likely. In other words, we need to compute (p ω |X) for each class $\omega_i$. However, the density functions provide only the likelihood of seeing a particular sample, given that the sample belongs to a specific class. i.e., the density functions

can be provided as p (X|ω) . The Bayesian rule provides us with an approach to compute the likelihood of the class for a given sample, from the density functions and related information.

# 2 Principle and Theory

The essence of the Bayesian approach is to provide a mathematical rule explaining how you should change your existing beliefs in the light of new evidence. In other words, it allows us to combine new data with their existing knowledge or expertise. The canonical example is to imagine that a precocious newborn observes his first sunset, and wonders whether the sun will rise again or not. He assigns equal prior probabilities to both possible outcomes, and represents this by placing one white and one black marble into a bag. The following day, when the sun rises, the child places another white marble in the bag. The probability that a marble plucked randomly from the bag will be white (i.e., the child's degree of belief in future sunrises) has thus gone from a half to two-thirds. After sunrise the next day, the child adds another white marble, and the probability (and thus the degree of belief) goes from two-thirds to three-quarters. And so on. Gradually, the initial belief that the sun is just as likely as not to rise each morning is modified to become a near-certainty that the sun will always rise.

In terms of classification, the Bayesian theorem allows us to combine prior probabilities, along with observed evidence to arrive at the posterior probability. More or less, conditional probabilities represent the probability of an event occurring given evidence.According to the Beyesian Theorem,if $P(\omega_i)$ , $P(X \mid \omega_i)$ , $\mathrm{i} = 1,2,\cdots,c$ ,and $X$ are known or given,the posterior probability can be derived as follows

$$P(\omega_i \mid X) = \frac{P(X \mid \omega_i)P(\omega_i)}{\sum_{j=1}^{c} P(X \mid \omega_j)P(\omega_j)} \qquad \text{i=1,…, c} \qquad (1)$$

Let the series of decision actions as $\{\mathrm{a}_1, \mathrm{a}_2, \cdots, \mathrm{a}_c\}$, the conditional risk of decision action $\mathrm{a}_\mathrm{i}$ can be computed by

$$R(a_i \mid X) = \sum_{j=1, j \neq i}^{c} \lambda(a_i, \omega_j) P(\omega_j \mid X), \qquad \text{i=1,…, c} \qquad (2)$$

Thus the minimum risk Bayesian decision can be found as

$$a_\mathrm{k}^* = \mathrm{Arg\ min_i}\ R(a_i \mid X), \qquad \text{i=1,…, c} \qquad (3)$$

## 3 Objective

The goals of the experiment are as follows:

(1) To understand the computation of likelihood of a class, given a sample.

(2) To understand the use of density/distribution functions to model a class.

(3) To understand the effect of prior probabilities in Bayesian classification.

(4) To understand how two (or more) density functions interact in the feature space to decide a decision boundary between classes.

(5) To understand how the decision boundary varies based on the nature of density functions.

## 4 Contents and Procedure

**Stage 1:**

(1) According to the above principle and theory in section 2.2, design a Bayesian classifier for the classification of two classes of patterns which are subjected to Gaussian normal distribution and compile the corresponding programme codes.

(2) In view of the normal cell class $\omega_1$ , the corresponding data of sample features are extracted as

$\Omega_1$ ={ -3.9847，-3.5549，-1.2401，-0.9780，-0.7932，-2.8531，-2.7605，-3.7287，-3.5414，-2.2692，-3.4549，-3.0752，-3.9934， -0.9780，-1.5799，-1.4885，-0.7431，-0.4221，-1.1186，-2.3462，-1.0826，-3.4196，-1.3193，-0.8367，-0.6579，-2.9683},

and the sample features of abnormal cell class $\omega_2$ are listed as

$\Omega_2$ = { 2.8792， 0.7932，1.1882，3.0682，4.2532，0.3271,0.9846,2.7648,2.6588}

The prior probabilities of both $\omega_1$ and $\omega_2$ are known as

$p(\omega_1) = 0.9, \qquad p(\omega_2) = 0.1$

The loss parameters for different decision action are given as table 1

Table 1 the loss parameters for different decision

| real class | | $\omega_1$ | $\omega_2$ | |
|---|---|---|---|---|
| decision action | $a_1$ | 0 | 1 | loss parameters |
| | $a_2$ | 6 | 0 | |

Suppose the conditional probability distributions are Gaussian, find the conditional probability density functions $p(X|\omega_1)$ and $p(X|\omega_2)$ and complete the design of Bayessian classifier with minimum risk, and then give a comparative analysis with the situation without considering decision loss.

Draw the curves of prior and posterior probability density functions, $p(X|\omega_1)$, $p(X|\omega_2)$, $p(\omega_1|X)$, and $p(X|\omega_2)$, give the classifying decision boundary function and illustration of classification result.

**Stage 2 ( towards an in depth study ):**

(1) Create a pattern dataset of multiple classes and high dimension with more than 50 samples for each class. Then design a Bayesian classifier and complete the corresponding experiments and comparative analysis by using self-supposed prior probabilities and loss parameters for the same terms as stage 1.

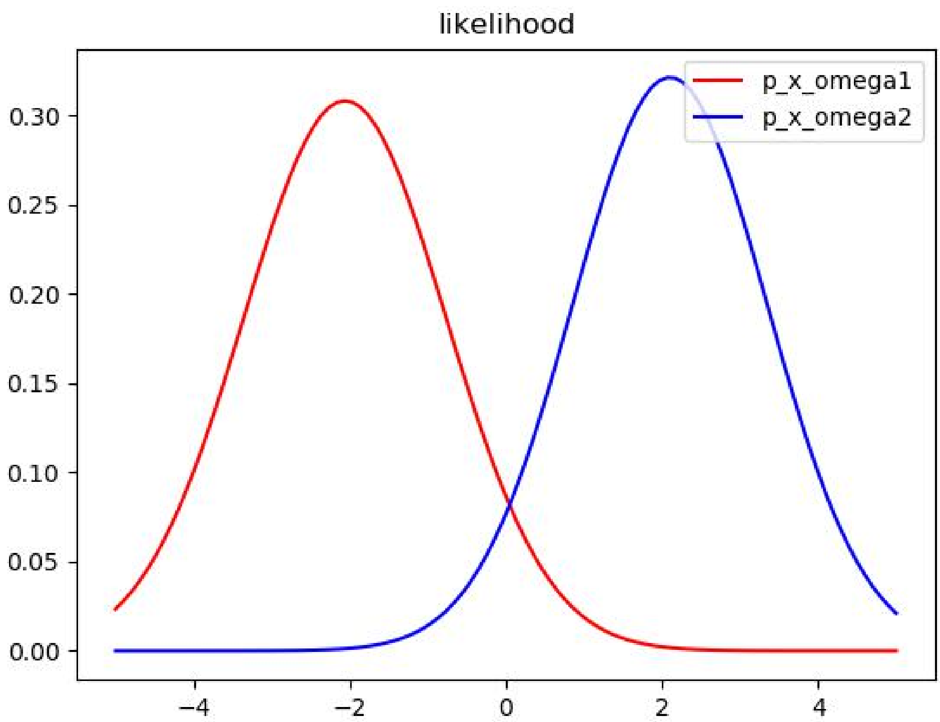
(2) Think and analyse the intrinsic relationship between the classifier of two classes and the one of multiple classes, give your comments.

**Stage 3：**

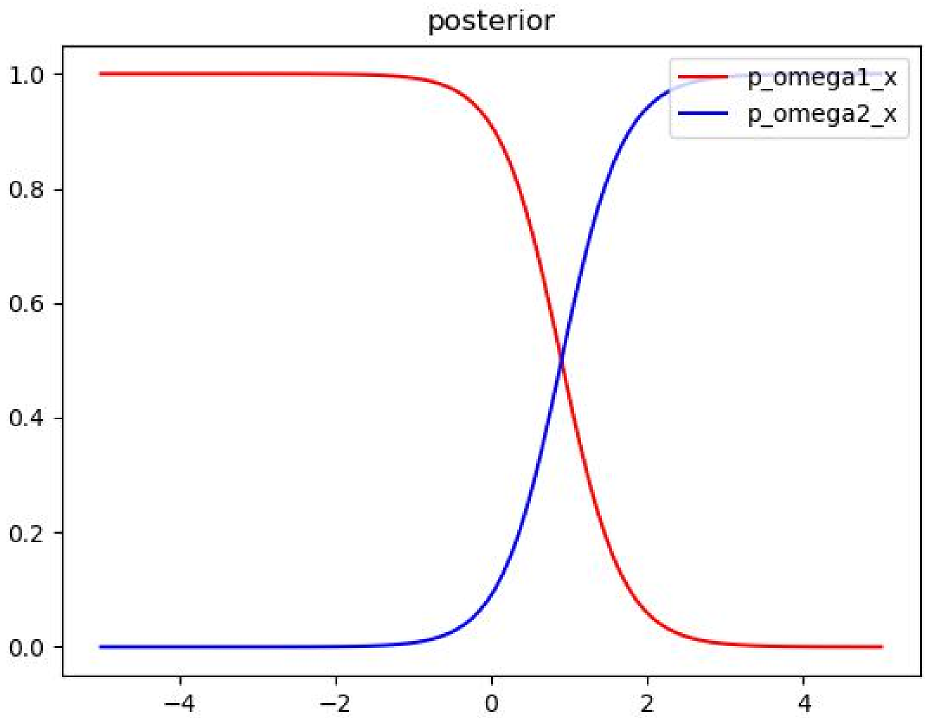
Explore the questions in the previous section and design experiments to answer these questions. Complete and submit an experiment report about all experiment results with comparative analysis and a summary of experiences about this experiment study.

## 5 Result and analyse

（1）Bayesian classifier without considering decision loss is a special situation of with minimum risk,when the risk parameters are 0/1.The likelihood picture are showed:




And the posterior are showed:




The decision boundary function is $R(a_1 | X) = R(a_2 | X)$ .If $R(a_1 | X) < R(a_2 | X)$ ,the classification result is that the new feature is classified to the first class $\omega_1$ .If $R(a_1 | X) > R(a_2 | X)$ ,the classification result is that the new feature is classified to the other class $\omega_2$ .

（2）Create a pattern dataset of 3 classes and 2 dimension with 100 samples for each class.The first dataset of class $\omega_1$ is distributed as Gaussion by mean(3,2) cov(2,2).The second dataset of class $\omega_2$ is distributed as Gaussion by mean(-2,2) cov(2,1).The third dataset of class $\omega_3$ is distributed as Gaussion by mean(0,-12) cov(1,1).The prior probabilities are 0.3, 0.2, 0.5 .The loss parameters are $\begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 3 \\ 2 & 2 & 0 \end{bmatrix}$ .The condition of decision $\forall j \in 1,2,3 \text{ but} \neq \text{i} \quad R(a_i | X) < R(a_j | X)$ will decide it belong to the ith class $\omega_i$ .For example,if you take the values of prediction sample is (3,2),then it will be decided as the first class $\omega_1$ ,if you take the values of prediction sample is (-2,2),then it will be decided as the first class $\omega_2$ ,and so on.You could run the python scripts I wrote to check result.

（3）The intrinsic relationship between the classifier of two classes and the one of multiple classes are the same.They both find the minimum of risk $R(a_i \mid X)$ to make a decision which class it belongs to.

All the code and instruction files are on my github.You can run them if you want:
https://github.com/Deep-Lan/Bayesian-Classifier.git