Two Category Classification Using Bayesian Decision Rule

Yadu Krishnan Sarathchandran

ysarathc@vols.utk.edu

Abstract

Bayesian decision theory is a statistical approach used in pattern recognition applications. This is useful when the probability distribution of the given data is understood to a certain degree. In this project, Bayesian decision rule is utilized to solve a synthetic two-class problem with two features, X and Y. The data is provided from Ripley's Pattern Recognition and Neural Networks website. We use the Maximum *A-priori* Probability (MAP) method to classify the data assuming a Gaussian distribution for the probability density. Maximum Likelihood Estimator (MLE) is used to estimate the parameters of the Gaussian function from the training data. These parameters are used in deriving the decision rules based on the MAP method. The accuracy of classification using each case (I, II and III) of the classifier is calculated to quantify the differences in decision making and the performance of the different classifiers. Accuracy is also tested by varying the prior probabilities from domain knowledge. A bimodal Gaussian probability is used to see how the classification efficiency changes.

1. Introduction

Machine learning (ML) is a scientific discipline where we let the computer perform a task by learning from previous data. It is considered or interpreted as a subset of Artificial Intelligence (AI). Applications and importance of machine learning are numerous. Starting from search results in a search engine such as Google, product suggestions in Amazon, friend suggestions in Facebook to self-driving cars and virtual-personal assistants like Alexa and Siri. Machine learning algorithms are classified into three main categories.

1. **Supervised Machine Learning**

   Supervised ML algorithms are based on a statistical model fitting the available, category labelled data called the 'training data'. The estimated parameters of the statistical model from the training data is then used to make decisions or predictions about a related, but unlabelled data (testing data) without explicitly programmed to do so [1]. An objective function is used in the prediction/decision making process. Discriminant function using Maximum *A-priori* Probability (MAP) is an example of an objective function based on the Bayesian decision rule in statistics/machine learning.

   In this project, we are dealing with a supervised machine learning example.

2. **Unsupervised Machine Learning**

   In unsupervised ML, also known as Clustering [2] we deal with a training data which is unlabeled and unclassified. In this case, the system forms clusters or groups of data. The algorithm is designed to act on the information given without any modeling using a statistical function [3]. Facial recognition algorithms and self-driving cars are examples of this type of ML.

3. **Reinforcement Learning**

As the name suggests, learning is reinforced in this type of ML. No desired label is given for the classification or prediction purposes. Based on the input data, an output is produced, then a feedback is given based on whether the prediction/decision is right or wrong. In other words, the feedback is binary, either 0 or 1. This process is analogous to a critic suggesting something is right or wrong in a performance, but doesn't reveal what exactly it is [2].

2. Objective

In this project, we deal with a synthetic Two-class Two-feature problem. The training data is taken from Ripley's Pattern Recognition and Neural Networks website. Both sets of data have two features, X and Y and two classes, 0 and 1. As an analogy, we can relate X to height and Y to weight of a person, and 0 to Male and 1 to Female classes. Thus, the training data gives information that what value of X, Y (Height, Weight) classifies a person as 0/1 (Male/Female). We plan to construct a decision rule based on which we classify the sample points in the testing data to the respective categories. The parameters of the Gaussian density are determined by using Maximum Likelihood Estimator. We were successful in deriving the three decision rules based using the MAP method to classify the testing data with great accuracy. The performance of the different classifier cases is analyzed based on their accuracy of classification. The effect of prior probabilities on the classification accuracy is also studied extensively. Finally, a bimodal Gaussian density is assumed instead of single-modal Gaussian to compare the performance.

3. Technical Approach

A Bayesian classifier approach is adopted for the classification purpose with the assumption that both training and testing data sets follow a Gaussian probability density function. The objective function used here is the discriminant function. Bayes' rule provides a way to calculate the posterior probability: $P\left(\frac{\omega_j}{x}\right) = \frac{p\left(\frac{x}{\omega_j}\right)P(\omega_j)}{p(x)}$, where $P\left(\frac{\omega_j}{x}\right)$ is the posterior probability, $p\left(\frac{x}{\omega_j}\right)$ is the conditional probability density function, $P(\omega_j)$ is the prior probability from domain knowledge and $p(x)$ is the normalization constant. Here, x is a column vector of dimension d, which is the number of features of the data. Based on MAP method, we classify the sample point to the class j if $P\left(\frac{\omega_j}{x}\right) > P\left(\frac{\omega_i}{x}\right)$. We use the log likelihood of $P\left(\frac{\omega_j}{x}\right)$, which is called the discriminant function $g_j(x)$. For our purpose, we use this function to construct the decision rule, strictly because log is a monotonically increasing function.

The Gaussian probability density is:

$$p(\vec{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\vec{x}-\vec{\mu})^T \Sigma^{-1}(\vec{x}-\vec{\mu})\right]$$

There are three cases for calculating the discriminant function for Gaussian density are explained below.

Case-1: The covariance matrices (Σ_i) of both classes are the same, which is also a diagonal and a scalar (σ^2) multiplication of the identity matrix of dimension 2, ($\Sigma_1 = \Sigma_2 = \sigma^2 I$). This means that the two features are independent of each other, also the variance in both features are same. In other words, 2-D Gaussian has same spreads along both axes.

Case-2: The covariance matrices of both classes are the same ($\Sigma_1 = \Sigma_2$), but possess non-zero non-diagonal elements. This implies that features are not independent of each other. The variances along both features are different implying the spread of the distribution along the two feature axes are different.

Case-3: The covariance matrices of the two classes are arbitrary in this case. This is the most general case of the discriminant function to classify data.

The decision boundary for each case is calculated by solving $g_1(x) - g_1(x) = 0$.

4. Experiments and Results

The training and testing data are imported using Pandas library in Python in Jupyter notebook as shown in Figure 1. The means and covariance matrices of both classes are estimated using a user-defined Gaussian function. The estimated means are given in Table 1 and depicted in Figure 1.

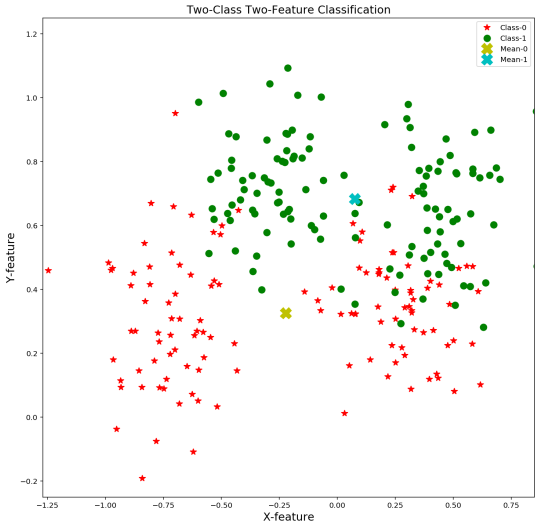

Figure 1 The imported training data.

Mean X	Mean Y	Class
-0.22147024	0.32575494	0
0.07595431	0.68296891	1

Table 1 Mean of two classes estimated from the training data

The Bayesian decision rules are derived by solving $g_1(x) - g_1(x) = 0$ to obtain the equations for three decision rules.

Case 1: $y = -0.83\,x + 0.44$
Case 2: $y = -0.14\,x + 0.5$

The decision boundaries for the three cases are depicted in Figure 2, along with the training data and their means.

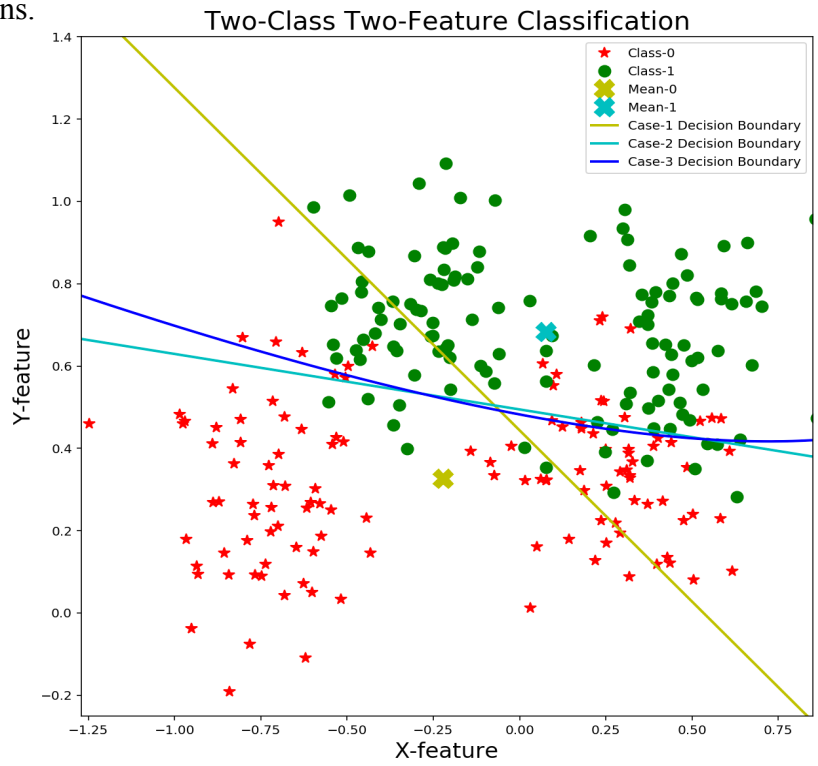

Figure 2 The decision boundaries for three cases along with the training data and mean of each class.

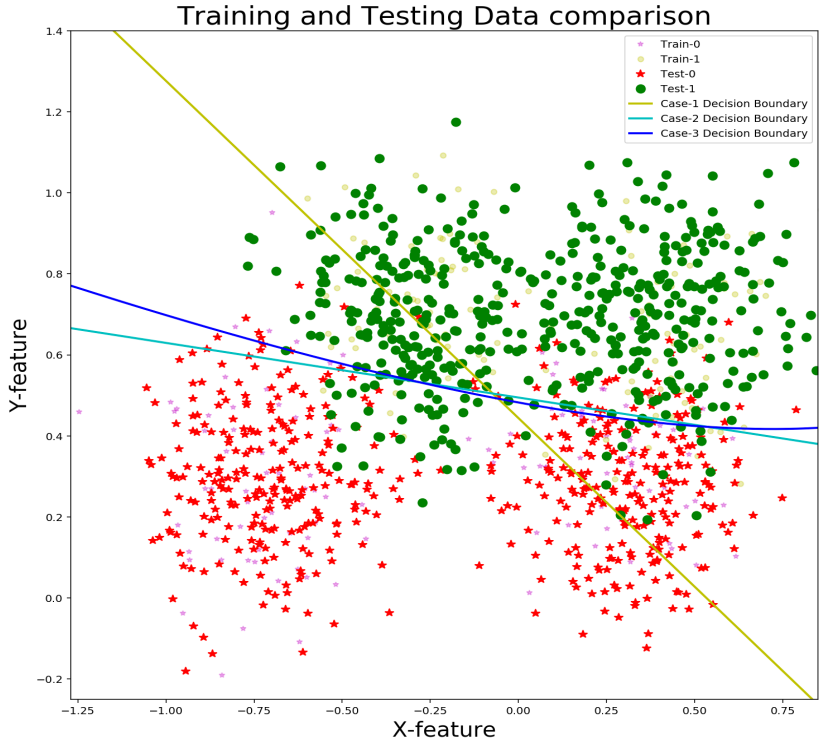

Figure 3 Visualization of the testing data in the backdrop of decision boundaries and the training data.

From the decision boundaries for all three cases, we can see that case-1 and case-2 are linear in nature and case-3 is quadratic. This can also be inferred visually from Figure 2 and 3. Also notice that in case-1 the decision boundary is basically equidistant from the sample means of both classes. case-2 and case-3 use the Mahalanobis distance for classification.

Performance Evaluation

The overall accuracy of classification is defined as below:

$$Accuracy = Number\ of\ Correct\ Predictions/Total\ Predictions$$

We analyze the performance of the classifiers by adjusting the prior probabilities. Priors are changed from 0.1 to 0.95 at an interval of 0.05. When the prior for class-0 is 0.1, the prior for class-1 is 0.9 and vice versa. In short, the sum of the two prior probabilities add up to 1. The overall accuracy is plotted against the prior probability value for class-0 to visualize the performance of each of the three cases of the Bayesian classifier. The maximum accuracy of each classifier occurs at a certain value of the prior probabilities. That is depicted as a red star on the Figure 4.

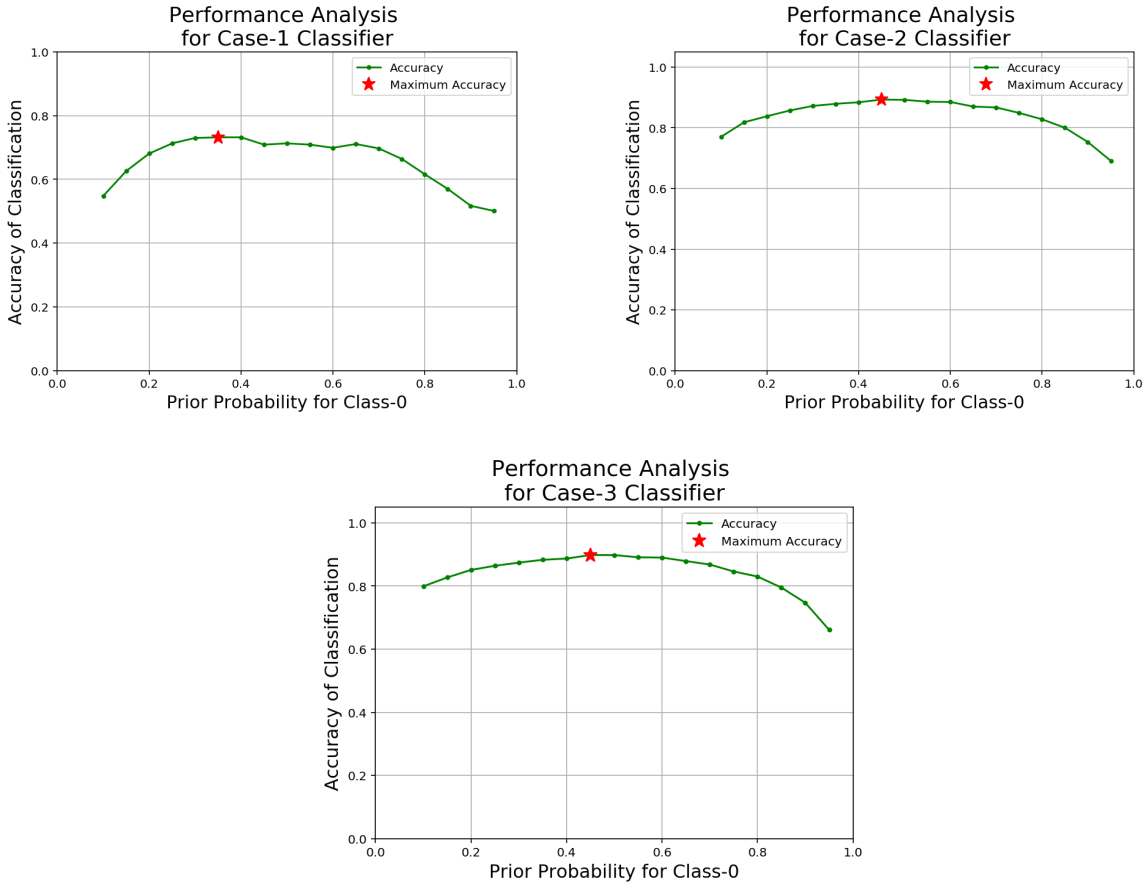

Figure 4 Visualization of the performance of the classifier based on overall accuracy

The class-wise accuracy of the classifier is defined as below:

$$Classwise\ Accuracy = Number\ of\ Correct\ Predictions\ in\ Class / Total\ Predictions\ in\ Class$$

The overall accuracies estimated for a prior probability of 0.5 are shown in Table 2 and class-wise accuracies for the same are shown in Table 3.

Classifier Case #	Overall Accuracy
1	0.713
2	0.892
3	0.898

Table 2 Overall accuracy for classifier cases

Classifier Case #	Class-0 Accuracy	Class-1 Accuracy
1	0.68	0.746
2	0.9	0.884
3	0.908	0.888

Table 3 Class-wise accuracy for classifier cases.

Bimodal Gaussian

Even though in all the classification procedure we used single modal Gaussian, from visual inspection of the training data, a bimodal Gaussian might be more appropriate to model the data and learn from it. It is simply the sum of two single-modal Gaussians.

In this scenario, we **only used case-3** of the Bayesian classifier. The means and covariance matrices for both classes has been estimated from the training data by visual inspection rather than by fitting the data, for the sake of convenience. Then, these parameters are used in classifying the same test data as in the previous cases to two classes.

The overall accuracy of the classification is estimated and found to be 0.832.

5. Discussion

We used a supervised machine learning algorithm to classify an unknown, unlabeled data set using a training data set and evaluated the performances of two special and one general cases of a Bayesian classifier with single-modal Gaussian probability density. MAP method was used in determining the decision boundaries using all cases and in classifying the test samples provided. From the results obtained, the highest overall accuracy belongs to the case-3 (general case), followed by case-2. The least accuracy was for case-1 at 0.713. This is expected because of the assumptions we made while doing the classifications. Higher the number of assumptions, lesser the accuracy. Class-wise accuracy also follows the same pattern as shown in Table 2 and 3. We can also infer from Table 2 and 3 that the difference in the classification accuracy of case-2 and case-3, 0.892 and 0.898 respectively, is negligible. Therefore, a cost –benefit analysis needs to be carried out before deciding between case-2 and case-3 for another classification purpose which require extensive memory storage and computational costs. We can also observe the dependence of accuracy on prior probabilities. Thus, a careful surveying of the domain is required before deciding the prior values. If there exists a disparity between the prior value for the highest accuracy of classification and the prior value obtained by surveying of the domain, careful

consideration needs to be taken. This scenario is a potential field of research to better the efficiency of unsupervised machine learning. Finally, we can see that the overall accuracy when using a bimodal Gaussian is better than case-1 of single-modal classification. Even though the overall accuracy of bimodal Gaussian doesn't exceed the case-2 and case-3 of the single-modal Gaussian, this is no reason to think that bimodal Gaussian is less efficient for our purpose. This is simply because we did not do a rigorous modeling of the training data using bimodal Gaussian compared to the other, instead we only did a visual inspection. Given that, an overall accuracy of 0.832 is impressive and suggests that the rigorous classification using a bimodal Gaussian will give a much better performance that a single-modal Gaussian.

6. References

1. Bishop, C. M. (2006), *Pattern Recognition and Machine Learning*, Springer.

2. R. O. Duda, P. E. Hart, D. G. Stork, *Pattern Classification*, 2nd Edition, John Wiley, 2001.

3. https://whatis.techtarget.com/definition/unsupervised-learning