Pattern Recognition

Classification

Student: Elena Mihalas

Abstract_ The aim of this report is to apply the basic probability relationships to equiprobable distributed classes in order to estimate the Bayesian error of the optimal decision rule and the system error for the training and the test data sets. Moreover, the report will contain the analysis and sketch of class distributions having the mean μ_i of class ω_i and covariance matrix Σ.

Introduction

All objects can be characterised by a specific pattern which describes its attributes. In most of the cases we want to extract a semantic meaning from numerous observation vectors of an object by grouping them in distribution classes which hold different characteristics of that object.

In machine learning, parametric classifiers are instructed on how to make an optimal decision upon the insertion of classes' boundary in the vector space once the prior probabilities and the class conditional likelihoods are known. Of course this decision rule takes into consideration the insertion of a decision cost related to the assignment of a pattern vector to a wrong class. In fact it is expected from the Gaussian classifiers to produce the minimum Bayes error by inserting each pattern vector in the class that shows the maximum "a posteriori probability" for that specific pattern vector. However, the system error sometimes is bigger than the Bayes error when the estimated means or variances are different from the true parameters of the analysed model.

This report will analyse various situations which produce a worse error estimation with respect to the minimum Bayes error. In particular, I will consider the case when the mean of one of the classes is not given and it is necessary to estimate the sample mean from a training set chosen from the distribution of the class. Furthermore, I will analyse the case when all the parameters has to be estimated from a test set and how these averaged values influence the estimation of the system error.

Sketch of class distributions

The Gaussian distribution is mainly used to represent the continuous probability distribution of random variables with a given mean and variance. For this report I am provided with the values of the mean μ_i for the class ω_i and with the unique covariance matrix Σ:

$$\mu_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \qquad \mu_2 = \begin{bmatrix} 4 \\ 4 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

In order to sketch the 2D isoprobable distribution curve, I computed the eigenvectors and the eigenvalues of my model by implementing the principal component analysis (PCA). The eigenvectors [u1, u2] are the directions in a 2D vector space where the data is spread out mostly. Each eigenvector has associated an eigenvalue σ_i^2 which indicates the variance of the data along the direction defined by the eigenvector. Thus, for my model, the estimated eigenvectors and eigenvalues are:

$$U = [u_1 \quad u_2] = \frac{\sqrt{2}}{2} * \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad V = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$$

By analysing the results, it can be seen that I obtained only one eigenvalue which means that the variance of the data along the second eigenvector is null. In fact the distributions collapse to one dimension and the two means have to be moved from the eigenmodel frame to the x and y frame. In order to obtain this transformation I had to multiply U by the actual means. Thus the new means' values, that I obtained, are **μ_1 = $[2\sqrt{2};\ 0]$** and **μ_2 =$4\sqrt{2};\ 0]$.** And if I want to represent the two distributions in 1D, I have to apply the following formula:

$$\mathrm{p(x} \mid \omega_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}}\, exp^{\frac{-(x-\mu_i)\hat{}2}{2\sigma_i^2}}$$ where **x** is the vector of observations (1)

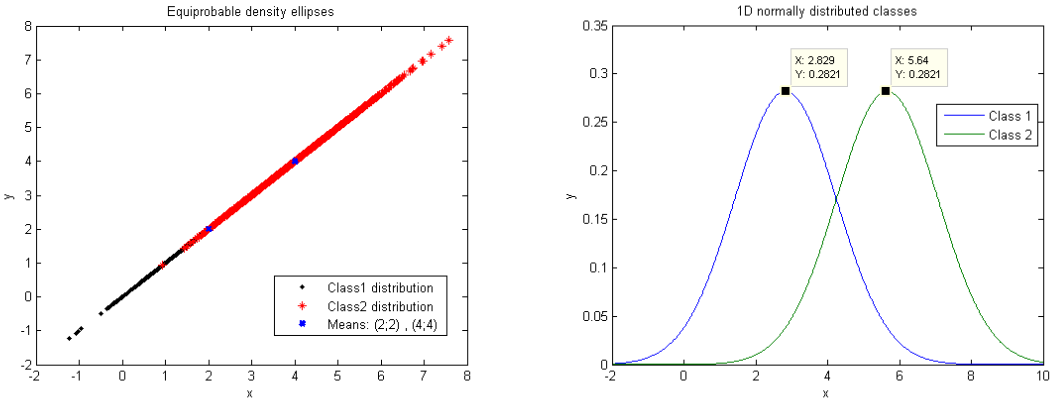

Figure 1_ The first graphic shows the distribution of the data in 2D along the direction [1; 1] with a standard deviation of $\sqrt{2}$. The second graphic shows the same information in 1D with vector **x** defined over 1000 samples chosen from the range [-2:10].

Bayesian Error

In Pattern Recognition is common to implement statistical models when the pattern vector *x* is made of unknown random variables, in order to evaluate the posterior probability P(ω|x) of an event given the prior probability P(ω). However, in my case I have to work upon two dependant class distributions, therefore I have to consider the probability of their joint occurrence, which is:

$$\mathrm{p}(\mathrm{x}, \omega_i) = \mathrm{p}(\mathrm{x} \mid \omega_i) * \mathrm{P}(\omega_i) = \mathrm{P}(\omega_i \mid \mathrm{x}) * \mathrm{p}(\mathrm{x})$$

Since the prior probabilities $P(\omega_1)$ and $P(\omega_2)$ are equal, I obtain two equiprobable normally distributed classes, and the posterior probabilities of my Gaussian distributions can be obtained by implementing the Bayes' Theorem:

$$P(\omega_i \mid x) = \frac{p(x|\omega_i) * P(\omega_i)}{p(x)}$$

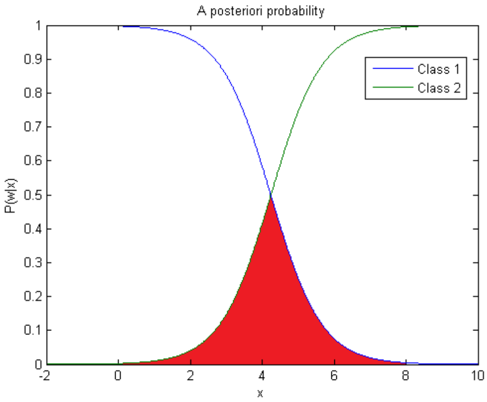

Figure 2_ The above graphic represents the "a posteriori probabilities" of the two class distributions, whereas the area coloured in red is the average error of optimal decision rule.

The "A posteriori probability" graph shows the trend of the two $P(\omega_i|x)$ and it can be noticed that while the class ω_1 probability is decreasing the class ω_2 probability is increasing. However, we must be aware that this is an ideal representation of the model because the distributions are built using the exact mean and variance values, which usually are not known in real situations.

As I have to estimate the Bayes error, I need first to evaluate the optimal decision rule for my model, which will assigns x to one of the classes based on the established boundary. Moreover, I have to partition the space, so that $x \epsilon \Gamma_j$, and to set the classifier to assign x -> ω_j only if the following statements are satisfied:

$$\sum_{i=1}^{m} \rho_{ij} P(\omega_i|x) = min_k \sum_{i=1}^{m} \rho_{ik} P(\omega_i|x)$$

The ρ_{ij} are the costs associated with the assignment of x to ω_j instead of $\omega_{i,}$, namely it is equal to 0 for correct decision (ρ_{ii}) and to 1 for incorrect decisions. By applying the zero-one costs assumption to the previous statement I obtain:

$$\sum_{i=1}^{m} \rho_{ik} P(\omega_i|x) = \sum_{i=1,\ i\neq k}^{m} 1 * P(\omega_i|x) = \sum_{i=1}^{m} 1 * P(\omega_i|x) - 1 * P(\omega_k|x) = 1 - P(\omega_k|x)$$

Therefore, the corresponding decision rule becomes:

Assign x -> ω_j if $P(\omega_j|x) = max_k P(\omega_k|x)$

At this point, I can evaluate the average error of the optimal decision rule, associated with the assignment of x to a wrong class, by computing the Bayes error:

$$\epsilon_B = 1 - \int max_j P(\omega_k|x)p(x)dx$$

Moreover, if I assume that S_i denotes the region where $P(\omega_i|\, x) = max_k\, P(\omega_k|\, x)$, I can express the Bayes error as class conditional error:

$$\epsilon_B \;= 1 - \sum_{i=1}^{2} \int_{S_i} p(x|\omega_i)P(\omega_i)\; dx$$

$$\epsilon_B \;= \sum_{i=1}^{2} P(\omega_i)\, [1 - \;\int_{S_i}\, p(x|\omega_i)dx]$$

The above formula is the formal definition of the sum of the area under each tail across the shared x range, namely the area coloured in red. In order to estimate the tail probability of a standard normal distribution, I had to solve the integral of a normal 1D distribution using the Q-function:

$$Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}}\, e^{\frac{-t\text{^}2}{2}} dt \;\rightarrow Q(x) = \frac{1}{2}\left(1 - erf\left(\frac{x}{\sqrt{2}}\right)\right)$$

Since my distribution has a mean and a variance, I evaluated the above formula for $\boldsymbol{x} = \frac{\boldsymbol{\theta - \mu_i}}{\boldsymbol{\sigma_i}}$**,** where **θ** is the average of the two means, namely it is the point on the x axis through which passes the vertical boundary of my model. Next I implemented the conditional Bayes error to the two probability regions S_i in order to compute the mean error of the class ω_i being assigned by mistake to class ω_j :

$$\epsilon_B = P(\omega_1) \int_{S_2} p(x|\omega_1)dx + P(\omega_2) \int_{S_1} p(x|\omega_2)dx = 0.5 * Q\left(\frac{4.243-2\sqrt{2}}{\sqrt{2}}\right) + 0.5 * Q\left(\frac{4.243-4\sqrt{2}}{\sqrt{2}}\right) = 0.157$$

Finally, I obtained a Bayes minimum error of $\epsilon_B \;= 0.157$ for my model with equal prior probabilities.

<u>Estimation of the decision rule and classifier's error</u>

At this point I have to derive the decision rule for the assignment of x to one of the two classes considering class ω_1's distribution slightly modified. In fact, I have to estimate the new sampled mean μ'_1 of N = 25 samples from class ω_1. Thus I have to apply the formulas for the estimation of the expected value and variance of an average extracted from "n" independent and identically distributed random variables.

The average is the value obtained by summing up the *n* random variables from a class distribution and by dividing the result by $\frac{1}{n}$, whereas the expected mean of the average is the expected mean of the sum multiply by a constant and the variance is the variance of the sum multiplied by the constant squared:

$$X^* = \frac{1}{n} * (X_1 + X_2 + \ldots + X_n) \quad\rightarrow\quad E(X^*) = \frac{1}{n} * E(X_1 + X_2 + \ldots + X_n) \quad\rightarrow\quad Var(X^*) = \left(\frac{1}{n}\right)^2 * Var(X_1 + X_2 + \ldots + X_n)$$

However, since the vector X's elements are independent, I can also apply the rule which says that the expected value of the sum is always the sum of the expected values and the variance of the sum is the sum of the variances:

$$E(X_1 + X_2 + \ldots + X_n) = E(X_1) + E(X_2) + \ldots + E(X_n) = \mu + \mu + \ldots + \mu = n*\mu$$

$$Var(X_1 + X_2 + \ldots + X_n) = Var(X_1) + Var(X_2) + \ldots + Var(X_n) = \sigma^2 + \sigma^2 + \ldots + \sigma^2 = n*\sigma^2$$

By combining together the two statements, I obtained that the expected mean of the average remains equal to the original mean of the distribution, whereas the variance decreases by a factor of $\frac{1}{n}$. Thus, the distribution of the mean tells us that 1/3 of the times the sample mean will be $\sqrt{\frac{\sigma^2}{n}}$ far from the true value of the mean.

Once I have defined μ_1' as μ_1 decreased by one standard deviation of the distribution of sample means ($\mu_1' = \mu_1 - \sqrt{\frac{\sigma^2}{n}}$), I determined the decision rule for the classifier using the new model with μ_2, Σ and μ_1'.

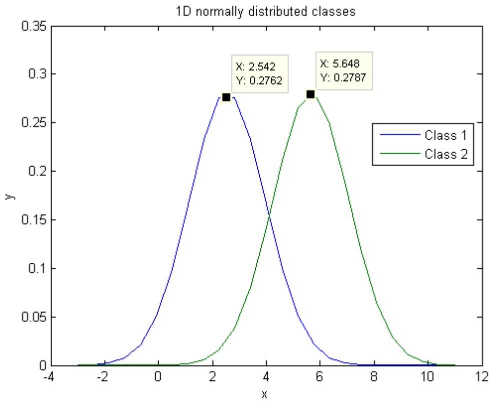

Figure 3_The graphic represents the two 1D normally distributed classes plotted over N = 25 samples from the range [-3, 11]. The blue parabola defines the class 1 with the mean **$\mu_1' = 2.546$** and variance $\sigma_2 = 2$, whereas the green parabola shows the distribution of class 2 with mean **$\mu_2 = 5.657$** and variance $\sigma_2 = 2$.

First, I established that the classifier had to assign x to ω_1 only if the next conditions were satisfied:

$$P(\omega_1 | x) = \max_j P(\omega_j | x)$$

$$p(x| \omega_1)*\mathrm{P}(\omega_1) = \max_j p(x| \omega_j)*\mathrm{P}(\omega_j)$$

Since my 2D model collapses to 1D distribution of probability, it was worth to simplify the above expression by defining p(x|ωi) as in eq.(1) and then taking the $\log_e$ of the entire expression:

$$\frac{-(x-\mu_1)^\wedge 2}{2\sigma_1^2} + log_e\mathrm{P}(\omega_1) > \frac{-(x-\mu_2)^\wedge 2}{2\sigma_2^2} + log_e\mathrm{P}(\omega_2)$$

As $\sigma_1 = \sigma_2 = \sigma$ and $\mathrm{P}(\omega_1) = \mathrm{P}(\omega_2)$, I simplified again the equation and obtained:

$$(x - \mu_1)^2 < (x - \mu_2)^2$$

The last equation is the expression for the computation of the nearest mean. Thus, by adopting the data provided, I obtained a Nearest Mean parametric decision rule for the Gaussian classifier.

Finally I computed the boundary by extracting the x value from the below equation:

$$x^2 - 2x\mu'_1 + \mu_1'^2 = x^2 - 2x\mu_2 + \mu_2^2$$

$$2x(\mu_2 - \mu_1') = \mu_2^2 - \mu_1'^2$$

$$2x(\mu_2 - \mu_1') = (\mu_2 - \mu_1')(\mu_2 + \mu_1')$$

$$x = \frac{(\mu_2 + \mu_1')}{2} = 4.1$$

Once obtained the exact position of the boundary, I sketched it on the original distribution in order to show that the decision boundary of the new model does not coincide with the original boundary, it is rather shifted to the left by a small amount (Fig.4). In other words it means that the designed classifier will on average estimate the membership of x to one of the two classes slightly worse with respect to the Bayesian optimal decision rule.

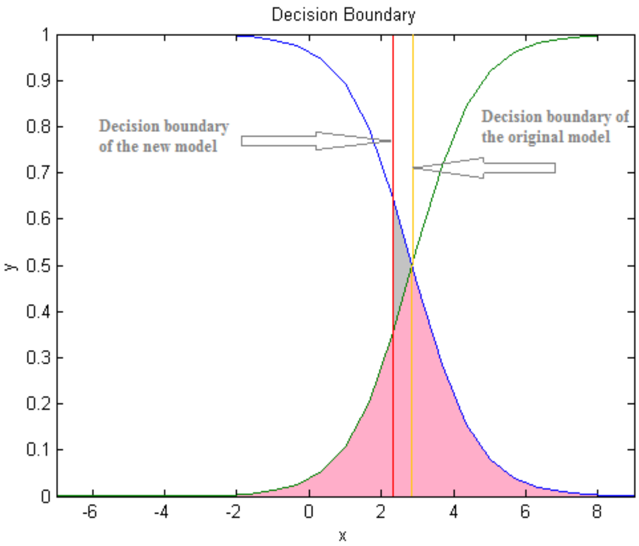

Figure 4_The graphic represents the trend of the two "a posteriori probabilities" and the decision boundaries sketched using the original means μ_i (yellow line) and the modified mean μ_1' (red line). It shows as well the average error area computed in Chapter 2 (pink region) and the amount of error (grey region) that is added to the Bayesian error after the modification of one of the means.

In order to prove the above affirmation, I computed the Q-function with $\boldsymbol{\theta} = \mathbf{4.1}$ and I obtained:

$$\mathrm{S_e} = 0.5 * \mathrm{Q}\left(\frac{4.1 - 2\sqrt{2}}{\sqrt{2}}\right) + 0.5 * \mathrm{Q}\left(\frac{4.1 - 4\sqrt{2}}{\sqrt{2}}\right) = 0.16$$

The classifier error is higher than 0.157. Obviously the difference between the errors is quite small because μ_1' is very close to the true mean of the distribution.

Evaluation of the classifier over a test set

In the previous section I designed a classifier by defining its decision rule. In this section, instead, I am going to test it over 25 samples taken from a normal distribution having covariance Σ and different means with respect to my original class distribution. In particular, I will assume that the mean of the test set from class ω₁ is μ_1' , whilst for the test set from class ω₂ the mean at the beginning is $\boldsymbol{\mu_2'} = \boldsymbol{\mu_2} + \sqrt{\frac{\sigma^2}{n}} = \mathbf{5.94}$ and afterwards it changes to $\boldsymbol{\mu_2''} = \boldsymbol{\mu_2} - \mathbf{2} * \sqrt{\frac{\sigma^2}{n}} = \mathbf{5.09}$

.

The independent test data set is given to the designed classifier in order to estimate the system's performance, which is mainly defined by the system error. Since, the parameters of the test set distribution are known, I derived the formula for the expected error of the system:

$$E_{\eta,x}\{\eta(x)\} = \int E_\eta\{\eta(x)\}p(x)dx$$

where η(x) is an error indicator: $\eta(x) = \begin{Bmatrix} 0 \text{ if x assigned to correct class} \rightarrow \text{error prob.} = 1 - \epsilon(x) \\ 1 \text{ if x assigned to incorrect class} \rightarrow \text{error prob.} = \epsilon(x) \end{Bmatrix}$

If I expand the above statement, by considering error indicator's real contribution, I obtain the expected error being equal to the actual error:

$$E_{\eta,x}\{\eta(x)\} = \int[1 * \epsilon(x) + 0 * (1 - \epsilon(x))]p(x)dx = \int \epsilon(x)p(x)dx = e$$

Note that ϵ(x) is the point wise error and it is defined as follows: $\epsilon(x) = 1 - max_i P(\omega_i|x)$

Therefore, the expected error of error indicator for my new statistical model, with means μ₁' and μ₂' or μ₂'' and decision boundary equal to the boundary found during the training session (θ=4.1), will be:

$$E_\eta\{\eta(x)\} = \mathrm{P}(\omega_1)\left[\int_{S_1} p(x|\omega_1)dx * 0 + \int_{S_2} p(x|\omega_1)dx * 1\right] + \mathrm{P}(\omega_2)\left[\int_{S_1} p(x|\omega_2)dx * 1 + \int_{S_2} p(x|\omega_2)dx * 0\right]$$

Case with μ₁' and μ₂': $E_\eta\{\eta(x)\} = 0.5 * \mathrm{Q}\left(\frac{4.1-2.546}{\sqrt{2}}\right) + 0.5 * \mathrm{Q}\left(\frac{4.1-5.94}{\sqrt{2}}\right) = 0.09$

Case with μ₁' and μ₂'': $E_\eta\{\eta(x)\} = 0.5 * \mathrm{Q}\left(\frac{4.1-2.546}{\sqrt{2}}\right) + 0.5 * \mathrm{Q}\left(\frac{4.1-5.09}{\sqrt{2}}\right) = 0.22$

At this point it would be interesting to know if the distribution of the expected error for the error estimate (e^) depends on the number N of the samples present in the data set. Thus the formula for the computation of the error estimate is: $e^{\wedge} = \frac{1}{N_{test}}\sum_{j=1}^{N_{test}} \eta(x_j)$

Since the expected value has a variance, it might be useful to express it in terms of error estimate as follows:

$$E_{\eta,x}\{(\eta(x) - e)^2\} = \int E_\eta\{\eta(x)^2\}p(x)dx - e^2 = \int \epsilon(x)p(x)dx - e^2 = e(1-e)$$

$$Var\ e^{\wedge} = E\{e^{\wedge}\} - e^2 = \frac{1}{N_{test}}\, e(1-e)$$

Thus, I obtain a variance which is inversely proportional to the number of samples in the data set. Thus, this relation between the two quantities suggests that it is necessary to consider large data sets in order to properly estimate the performance of the system because the variance of the expected error estimate decreases with the increase of N_{test}.

Conclusions

The aim of this report was to design a Gaussian classifier and to compute the system error for the training and the test sessions. Furthermore it was necessary to evaluate the classifier's performance by comparing the obtained errors with the Bayes minimum error, which represents the lower bound achievable for a given classification.

It was showed that each dataset was built using a sampled mean of 25 random values taken from the original class distribution. The sampling process generated each time slightly different dataset distributions based on the variance of the expected mean. Nevertheless, in the report was proven that this disadvantage can be overcome by increasing the number of samples which in turn decreases the variation of the sampled mean and therefore it is more probable to obtain a distribution similar to the original model's distribution.

Thus, by training my system on only 25 samples, I obtained a system error during the training session slightly higher (0.16) than the Bayes minimum error (0.157). This result, however, was expected because the sampled mean shifted to the left the class 1 distribution by a small amount and accordingly the boundary changed location, generating an unbalanced assignment of pattern vector x to each of the two classes.

On the contrary, the system error obtained from the test session showed both worse (0.22) and better (0.09) results than the Bayes minimum error. In fact, it indicates that the system error extracted from the test session is not reliable for evaluating the classifier's performance unless the number of samples is large, which will bring the system error closer to the actual error of the model. Otherwise the obtained error will vary considerably because all the parameters of the test set distribution are unknown and hence guessed.