Open your mind. LUT.
Lappeenranta University of Technology
LUT Machine Vision and Pattern Recognition 2015-10-02

BM40A0700 Pattern Recognition

Lasse Lensu

Exercise 5: Bayesian classification

1. Effect of unequal variances (2 *bonus* points): Consider two normal probability distributions

$$p(x|\omega_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left[-\frac{1}{2}\left(\frac{x-\mu_i}{\sigma_i}\right)^2\right] \tag{1}$$

   with **equal** a priori probabilities $P(\omega_1) = P(\omega_2)$ and with **unequal** standard deviations (STDs) $\sigma_1 = 1$ and $\sigma_2 = 2$. Determine a minimum-error classifier, derive the natural logarithm form of the discriminant function and use that to determine the decision boundary for the classifier in the case $\mu_1 = 3$ and $\mu_2 = 6$.

   Calculate the decision boundary by hand, and verify by plotting the distributions in Matlab.

   Notes: The calculation by hand may get complicated.

2. Classification in two dimensions with unequal covariances (2 points): Consider the following three-class classification problem. Each class is normally distributed with equal a priori probabilities, and the following covariances and means:

$$\Sigma_1 = \begin{pmatrix} 1/2 & 0 \\ 0 & 2 \end{pmatrix} \quad \Sigma_2 = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \quad \Sigma_3 = \begin{pmatrix} 2 & 0 \\ 0 & 1/2 \end{pmatrix} \tag{2}$$

$$\mu_1 = \begin{pmatrix} 3 \\ 6 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} 3 \\ -2 \end{pmatrix} \quad \mu_3 = \begin{pmatrix} 11 \\ -2 \end{pmatrix} \tag{3}$$

   Calculate the decision boundaries, and draw a plot that shows the class centers and decision boundaries.

   Hints:

   (a) The decision boundary between the first and second class is a parabola which opens to the direction of positive x_2-axis (x_2 a second degree polynomial of x_1).

   (b) The decision boundary between the second and third class is a parabola which opens to the direction of positive x_1-axis (x_1 a second degree polynomial of x_2).

   (c) The decision boundary between the first and third class is linear, but solving it requires solving a second degree equation of x_2, which contains x_1 terms.

   (d) If solving the decision boundaries mathematically seems too complicated, you may alternatively compute the discriminant function values and visualize the decision boundaries, for example, by creating a meshgrid of points, and plotting the points with different colors indicating which class a sample at that location would be classified into.

3. Statistical classifier with normal distribution (2 points): Construct a Matlab function

```
function C = bayes1(mu, Sigma, data)
```

PO BOX 20, FI-53851 LAPPEENRANTA, FINLAND
E-mail: lasse.lensu@lut.fi

that performs Bayesian classification using a Gaussian distribution to model class conditional probabilities. The function parameters and the output are as follows:

- `mu` is a matrix of means so that each column represents a single class and each row is a particular attribute.
- `Sigma` is the three-dimensional array of covariance matrices, where the covariance matrices are along the third dimension.
- `data` is the matrix of data to classify, each sample in a single column.
- `C` is a vector of classes which should include one value for each column in `data` with values ranging from one to the number of columns in `mu` (that is, the number of classes).

A priori probabilities are considered to be unknown and thus should not be taken into account.

For example, the classifier can perform as follows: A $c \times m$ matrix can be constructed where c is the number of classes and m is the number of samples (data). Element (i, j) of the matrix should then contain the value of the ith discriminant function with the sample j. The class of sample j can then be found as the index i of the column j where the value is maximal.

Test your classifier using the given data. Plot your classification result and the correct classes.

Notes: This problem cannot be solved without errors using this data and Gaussian distribution model.

Hints:

(a) You can refer to the covariance matrix of the ith class in Matlab by `Sigma(:, :, i)`. One way to get the number of classes is by `numofclasses = size(mu, 2);`. You will perhaps also need `length`, `size`, `max`, and `inv` commands. The discriminant functions have been presented in the lecture notes.

(b) To visualise, first load the data into Matlab, plot the classes using `plotclass(mu, Sigma);` and the data using `plot(data(1, :), data(2, :), 'o');`. You can also plot a single class using (for the first class) `plot(data(1, classes == 1), data(2, classes == 1), 'rx');`

(c) Comment your code so someone else can understand it.

Additional files: `bayes1data.mat, plotclass.m`.