

IN 5520 2019

Classification 2

Exercises related to the lecture on 16.10.19

You will need your source code for Exercise 1 when doing Mandatory Exercise part 2, so implement this now 😊

Exercise 1. Matlab exercise for classification based on a multivariate Gaussian classifier.

Step 1: Implement a Gaussian classifier using a d-dimensional feature vector

For the algorithm, see lecture foils. It is recommended that you use Matlab built-in functions for matrix inversion and computing the determinant, but write the remaining algorithm yourself. (You can use built-in functions `mean()` and `cov()`)

Step 2: Train the classifier

Train the classifier on the image `tm_train.png` using the 4 classes defined in the mask file `tm_train.png`. Estimates of the class-specific mean vector and covariance matrix are found on the lecture foils. You may use Matlab functions `mean()` and `cov()` here.

If you want to verify that your code gives the correct classification labels, check the resulting classification image you get when you classify the entire image `tm.mat` with the classification we produced (`tm_classres.mat`). In this image each pixel is assigned class labels 1-4.

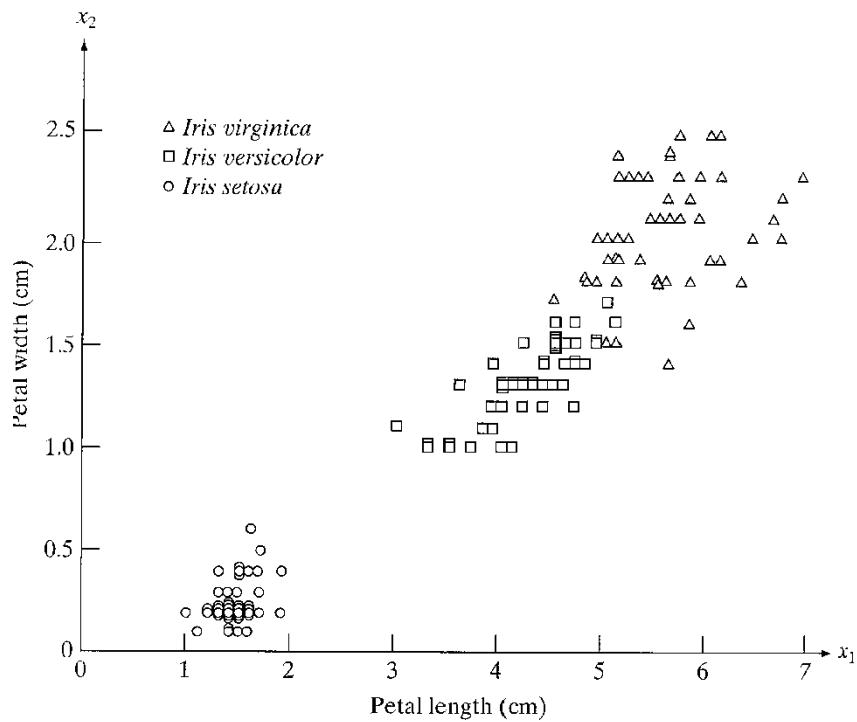
Step 3: Find the classification accuracy for classification using all features.

Run the classification on the multivariate (6-band) input image `tm.mat`. Compute the percentage of correctly classified pixels when using all features, and compare it to using single features. The classification accuracy should be computed on the test image `tm_test.png`.

Also try the simplified covariance matrix $\Sigma = \sigma^2 I$. Which version gives the highest classification accuracy?

2. Finding the decision functions for a minimum distance classifier.

A classifier that uses equal diagonal covariance matrices is often called a minimum distance classifier, because a pattern is classified to class that is closest when distance is computed using Euclidean distance. See plot on next page.



- In the above figure, find the class means just by looking at the plot.
- If this data is classified using a minimum distance classifier, sketch the decision boundaries on the plot.

3. Discriminant functions

A classifier that uses Euclidean distance computes distance from pattern \mathbf{x} to class j as:

$$D_j(x) = \|\mathbf{x} - \mu_j\|$$

Show that classification with this rule is equivalent to using the discriminant function

$$d_j(x) = x^T \mu_j - \frac{1}{2} \mu_j^T \mu_j$$

4. Exercise 3

In a three-class two-dimensional problem the feature vectors in each class are normally distributed with covariance matrix

$$\Sigma = \begin{bmatrix} 1.2 & 0.4 \\ 0.4 & 1.8 \end{bmatrix}$$

The mean vectors for the three classes are $[0.1, 0.1]^T$, $[2.1, 1.9]^T$, $[-1.5, 2.0]^T$.

Assuming that the classes are equally probable:

- Classify the feature vector $[1.6, 1.5]^T$ according to a Bayesian classifier with the given covariance matrix.
- Draw the curves of equal Mahalanobis distance from the class with mean $[2.1, 1.9]^T$.

5. Exercise 4

Given a two-class classification problem:

- $P(\omega_1) = P(\omega_2)$
- $p(x|\omega_1) = N(\mu_1, \Sigma)$, $p(x|\omega_2) = N(\mu_2, \Sigma)$
- $\mu_1 = [0 \ 0]$; $\mu_2 = [3 \ 3]$; $\Sigma = \begin{bmatrix} 1.1 & 0.3 \\ 0.3 & 1.9 \end{bmatrix}$, $\Sigma^{-1} = \begin{bmatrix} 0.95 & -0.15 \\ -0.15 & 0.55 \end{bmatrix}$

Classify feature vector $x = [1.0 \ 2.2]$ using Bayesian classification.