

Exercise 6b - Advanced segmentation. Fisherman's Linear discriminant analysis for segmentation

Introduction

The exercise is to the extent of the pixel-wise classification problem from being based on the intensity histogram of a single image modality to combining multiple image modalities. Hence, here we wish to segment image features into two classes by training a classifier based on the intensity information from multiple image modalities. Multiple-image modalities just mean a series of images that contain different but complementary image information of the same object. It is assumed that the image modalities have the same size, so we have a pixel-to-pixel correspondence between the two images. An image feature is an identifiable object in the image e.g., of a dog, moon rocket, or brain tissue types that we wish to segment into individual classes.

Here we aim to segment two types of brain tissues into two feature classes. To improve the segmentation, we wish to combine two MRI image modalities instead of a single: one is a "T1 weighted MRI" and the other is a "T2 weighted MRI". Both are acquired at the same time.

Exercise - You simply go step-by-step and fill in the command lines and answer/discuss the questions (Q1-Q12).

Theory

The Linear Discriminate Classifier

As a classifier, we will use a class of linear discriminate functions that aims to place a hyperplane in the multi-dimensional feature space that acts as a decision boundary to segment two features into classes. Since we only look at image intensities of two image modalities our multi-dimensional feature space is a 2D intensity histogram. The linear discriminant classifier is based on the Bayesian theory where the posterior probability is the probability of voxel x belonging to class C_i . The voxel x belongs to the class with the highest posterior probability.

You can find an **important** description of the theory behind LDA in - Exercise theory

Learning Objectives

1. Implement, train and evaluate multi-dimensional segmentation using a Linear Discriminate classifier i.e. Fisherman's Linear discriminant analysis
2. To visualise the 1D intensity histograms of two different image modalities that contain different intensity information of the same image features.

3. To identify the expected intensity thresholds in each of the 1D histograms that best segment the same feature in the two image modalities.
4. To visually the 2D histogram of two image modalities that map the same object but with different intensity information.
5. To interpret the 2D histogram information by identifying clusters of 2D intensity distributions and relate these to features in the images.
6. To draw an expected linear hyper plane in the 2D histogram that best segment and feature in the two image modalities
7. To extract training data sets and their corresponding class labels from expert drawn regions-of-interest data, and map their corresponding 2D histogram for visual inspection
8. To relate the Bayesian theory to a linear discriminate analysis classifier for estimating class probabilities of segmented features.
9. To judge if the estimated linear or a non-linear hyper plane is optimal placed for robust segmentation of two classes.

Installing Python packages

In this exercise, we will be using numpy, scipy and scikit-image. You should have these libraries installed, else instructions can be found in the previous exercises.

We will use the virtual environment from the previous exercise (course02502).

Exercise data and material

The data and material needed for this exercise can be found here:

- Exercise data

ex6_ImgData2Load.mat contains all image and ROI data which are loaded into the variables:

- **ImgT1** One axial slice of brain using T1W MRI acquisition
- **ImgT2** One axial slice of brain using T2W MRI acquisition
- **ROI_WM** Binary training data for class 1: Expert identification of voxels belonging to tissue type: White Matter
- **ROI_GM** Binary training data for class 2: Expert identification of voxels belonging to tissue type: Grey Matter

LDA.py A python function that realise the Fisher's linear discriminant analyse as described in Note for the lecture.

Image Segmentation

Start by importing some useful functions:

```
import numpy as np
import matplotlib.pyplot as plt
```

```
import scipy.io as sio
```

And the data:

```
in_dir = 'data/'
in_file = 'ex6_ImagData2Load.mat'
data = sio.loadmat(in_dir + in_file)
ImgT1 = data['ImgT1']
ImgT2 = data['ImgT2']
ROI_GM = data['ROI_GM'].astype(bool)
ROI_WM = data['ROI_WM'].astype(bool)
```

Exercise 1

Display both the T1 and T2 images, their 1 and 2D histograms and scatter plots. Tips: Use the `plt.imshow()`, `plt.hist()`, `plt.hist2d()` and `plt.scatter()` functions. Add relevant title and label for each axis. One can use `plt.subplots()` to show more subfigures in the same figure. Remove intensities from background voxels for 1D and 2D histograms.

`imshow` image coordinates

The two MRI image modalities contain different types of intensity classes:

1. (Orange): The White Matter (WM) is the tissue type that contain the brain network - like the cables in the internet. The WM ensure the communication flow between functional brain regions.
2. (Yellow): The Grey Matter (GM) is the tissue type that contain the cell bodies at the end of the brain network and are the functional units in the brain. The functional units are like CPUs in the computer. They are processing our sensorial input and are determining a reacting to these. It could be to start running.
3. (Magenta): Cerebrospinal fluid (CSF) which is the water in the brain
4. (Green): Background of the image

Q1: What is the intensity threshold that can separate the GM and WM classes (roughly) from the 1D histograms?

Q2: Can the GM and WM intensity classes be observed in the 2D histogram and scatter plot?

Exercise 2

Place trainings examples i.e. ROI_WM and ROI_GM into variables C1 and C2 representing class 1 and class 2 respectively. Show in a figure the manually expert drawings of the C1 and C2 training examples.

Tips: use `plt.imshow()`

Q3: Does the ROI drawings look like what you expect from an expert?

Exercise 3

For each binary training ROI find the corresponding training examples in `ImgT1` and `ImgT2`. Later these will be extracted for LDA training.

Tips: If you are a MATLAB-like programming lover, you may use the `np.argwhere()` function appropriately to return the index to voxels in the image full filling e.g. intensity values >0 hence belong to a given class. Name the index variables `qC1` and `qC2`, respectively.

Q4: What is the difference between the 1D histogram of the training examples and the 1D histogram of the whole image? Is the difference expected?

Exercise 4

Make a training data vector (`X`) and target class vector (`T`) as input for the `LDA()` function. `T` and `X` should have the same length of data points.

X: Training data vector should first include all data points for class 1 and then the data points for class 2. Data points are the two input features `ImgT1`, `ImgT2`

T: Target class identifier for `X` where '0' are Class 1 and a '1' is Class 2.

Tip: Read the documentation of the provided LDA function to understand the expected input dimensions.

Exercise 5

Make a scatter plot of the training points of the two input features for class 1 and class 2 as green and black circles, respectively. Add relevant title and labels to axis

Q5: How does the class separation appear in the 2D scatter plot compared with 1D histogram. Is it better?

Exercise 6

Train the linear discriminant classifier using the Fisher discriminant function and estimate the weight-vector coefficient `W` (i.e. w_0 and w) for classification given `X` and `T` by using the `W=LDA()` function. The LDA function outputs `W=[[w01, w1]; [w02, w2]]` for class 1 and 2 respectively.

Tip: Read the Bishop note on Chapter 4.

`W = LDA(X,T)`

Exercise 7

Apply the linear discriminant classifier i.e. perform multi-modal classification using the trained weight-vector `W` for each class: It calculates the linear score

Y for **all** image data points within the brain slice i.e. $y(x) = w + w_0$. Actually, $y(x)$ is the $\log(P(C_i|x))$.

```
Xall= np.c_[ImgT1.ravel(), ImgT2.ravel()]
Y = np.c_[np.ones((len(Xall), 1)), Xall] @ W.T
```

Exercise 8

Perform multi-modal classification: Calculate the posterior probability i.e. $P(C_1|X)$ of a data point belonging to class 1

Note: Using Bayes [Eq 1]: Since $y(x)$ is the log of the posterior probability [Eq2] we take $\exp(y(x))$ to get $P(C_1|X) = P(X|\mu, \sigma)P(C_1)$ and divide with the marginal probability $P(X)$ as normalisation factor.

```
PosteriorProb = np.clip(np.exp(Y) / np.sum(np.exp(Y), 1)[: , np.newaxis]), 0, 1)
```

Exercise 9

Apply segmentation: Find all voxles in the T1w and T2w image with $P(C_1|X) > 0.5$ as belonging to Class 1. You may use the `np.where()` function. Similarly, find all voxels belonging to class 2.

Exercise 10

Show scatter plot of segmentation results as in 5.

Q6 Can you identify where the hyperplane is placed i.e. $y(x)=0$?

Q7 Is the linear hyper plane positioned as you expected or would a non-linear hyper plane perform better?

Q8 Would segmentation be as good as using a single image modality using thresholding?

Q9 From the scatter plot does the segmentation results make sense? Are the two tissue types segmented correctly.

Exercise 11

Q10 Are the training examples representative for the segmentation results? Are you surprised that so few training examples perform so well? Do you need to be an anatomical expert to draw these?

Q11 Compare the segmentation results with the original image. Are the segmentation results satisfactory? Why not?

Q12 Is one class completely wrong segmented? What is the problem?