27/04/2018

Pattern Recognition- Homework 1

Author: Ali Berrada

Program: MAIA 2017/19

Notes

The scripts for this homework were written in Octave but should be compatible with MATLAB.

The script for “Problem 1.5” is attached as a file while the scripts used for the other questions can be found in the GitHub repository:

https://github.com/ali-yar/maia-patternrecognition

Problem 1.1

Breast cancer diagnosis is a common field where pattern recognition is used. Early and accurate diagnosis is very crucial to prevent the increasing risk of death while today's CAD systems exceeds the accuracy rate of the trained medical staff.

Among the features taken into consideration are clump thickness, bare nuclei, mitoses and uniformity of cell size and shape*.

The decision by the system is to classify the sample as benign or malignant. Neural Networks are commonly used for this classification.

There are several types and variants for Neural Networks like BP, ELM and Convolutional, but we can use a general description where we have a set of neurons in input, hidden and output layers that perform certain computations. We also need to have training, validation and test sets. The training set is used to train the network which gradually adjust its neurons' weights improving by that the results on the validation set. The test set will be used to measure the accuracy of the network through the confusion matrix. If the network achieves results lower than desired, it undergoes more training.

* W. H. Wolberg and O. L. Mangasarian, "Multisurface method of pattern separation for medical diagnosis applied to breast cytology," in Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp. 9193–9196.

Problem 1.2

a) The conditional densities have the form of a univariate Gaussian density.

So: $p(x|w_1) = k_1 \cdot \exp(-\frac{x^2}{15}) = \frac{1}{\sqrt{2\pi}\sigma} \cdot \exp(-\frac{(x-\mu_1)^2}{2\sigma^2})$

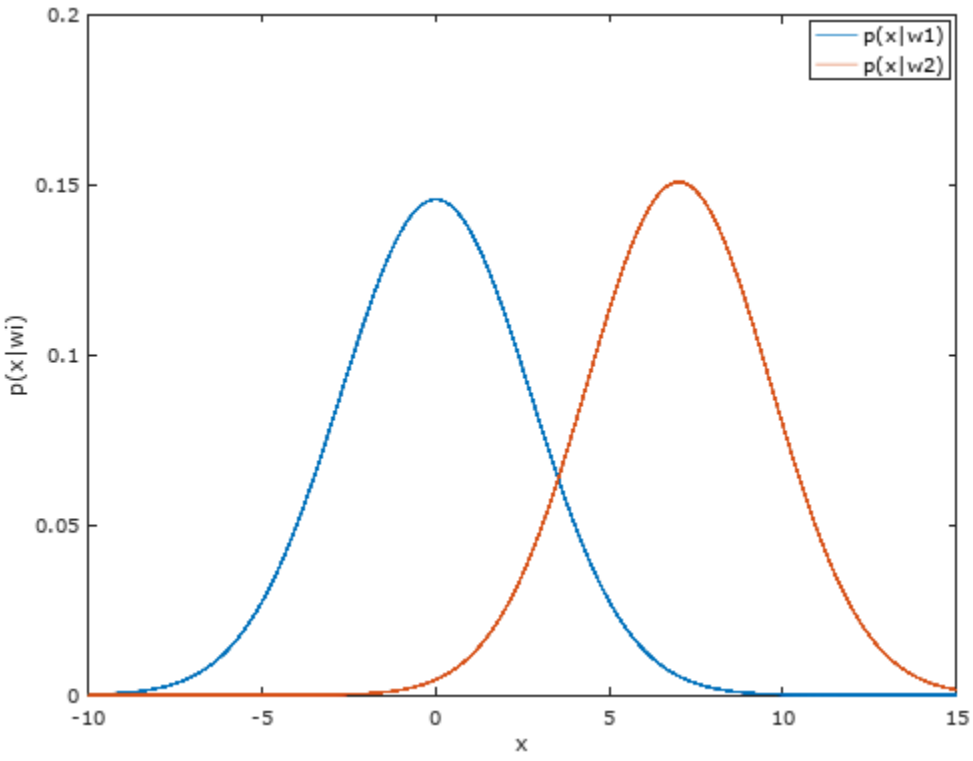
$\Rightarrow -\frac{x^2}{15} = -\frac{(x-\mu_1)^2}{2\sigma^2}$ and $k_1 = \frac{1}{\sqrt{2\pi}\sigma}$

$\Rightarrow \sigma = \sqrt{\frac{15}{2}}, \mu_1 = 0$ and $k_1 = \frac{1}{\sqrt{15\pi}}$

and: $p(x|w_2) = k_2 \cdot \exp(-\frac{(x-7)^2}{14}) = \frac{1}{\sqrt{2\pi}\sigma} \cdot \exp(-\frac{(x-\mu_2)^2}{2\sigma^2})$

$\Rightarrow -\frac{(x-7)^2}{14} = -\frac{(x-\mu_2)^2}{2\sigma^2}$ and $k_2 = \frac{1}{\sqrt{2\pi}\sigma}$

$\Rightarrow \sigma = \sqrt{7}, \mu_2 = 7$ and $k_2 = \frac{1}{\sqrt{14\pi}}$



b) Let's represent in terms of discriminant functions:

$g_1(x) = \ln p(x|w_1) + \ln P(w_1)$ and $g_2(x) = \ln p(x|w_2) + \ln P(w_2)$

The equation of the decision boundary is then:

$g_1(x) - g_2(x) = 0$

$\Rightarrow \quad \ln p(x|w_1) - \ln p(x|w_2) + \ln P(w_1) - \ln P(w_2) = 0$

$\Rightarrow \quad \ln(\frac{k_1}{k_2}) - \frac{x^2}{15} + \frac{(x-7)^2}{14} - \ln(\frac{P(w_2)}{P(w_1)}) = 0$

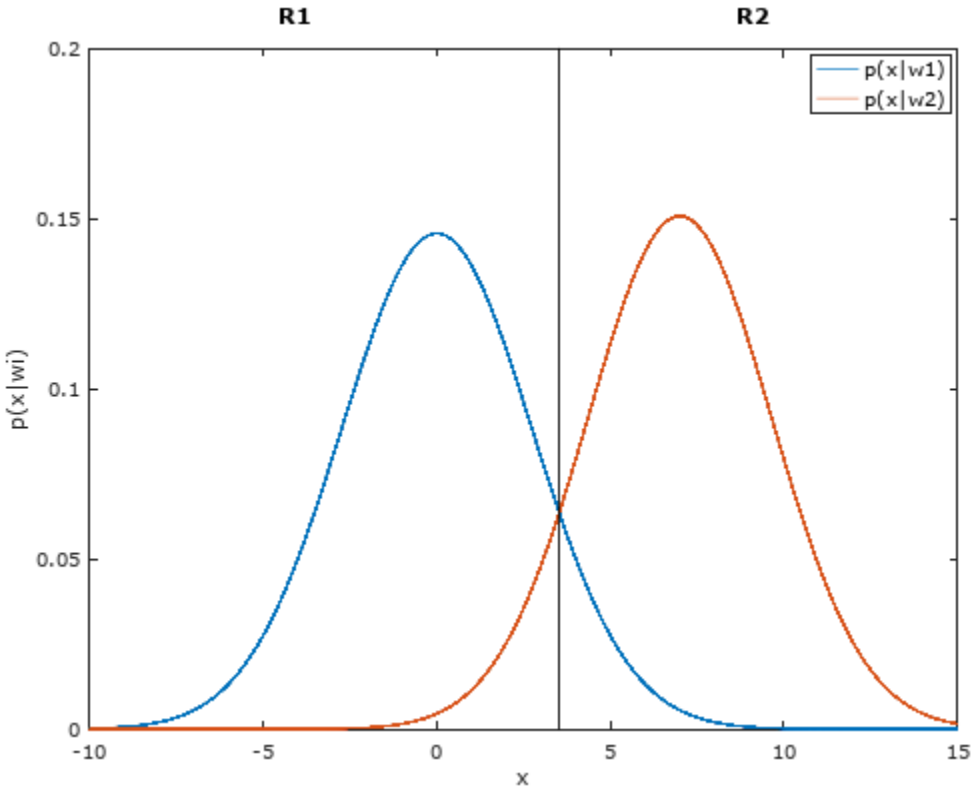
$\Rightarrow \quad x^2 - 210x + 735 + 210.\ln\left(\frac{k_1}{k_2}\right) - 210.\ln\left(\frac{P(w_2)}{P(w_1)}\right) = 0$

$\Rightarrow \quad x^2 - 210x + 727.76 - 210.\ln\left(\frac{P(w_2)}{P(w_1)}\right) = 0$

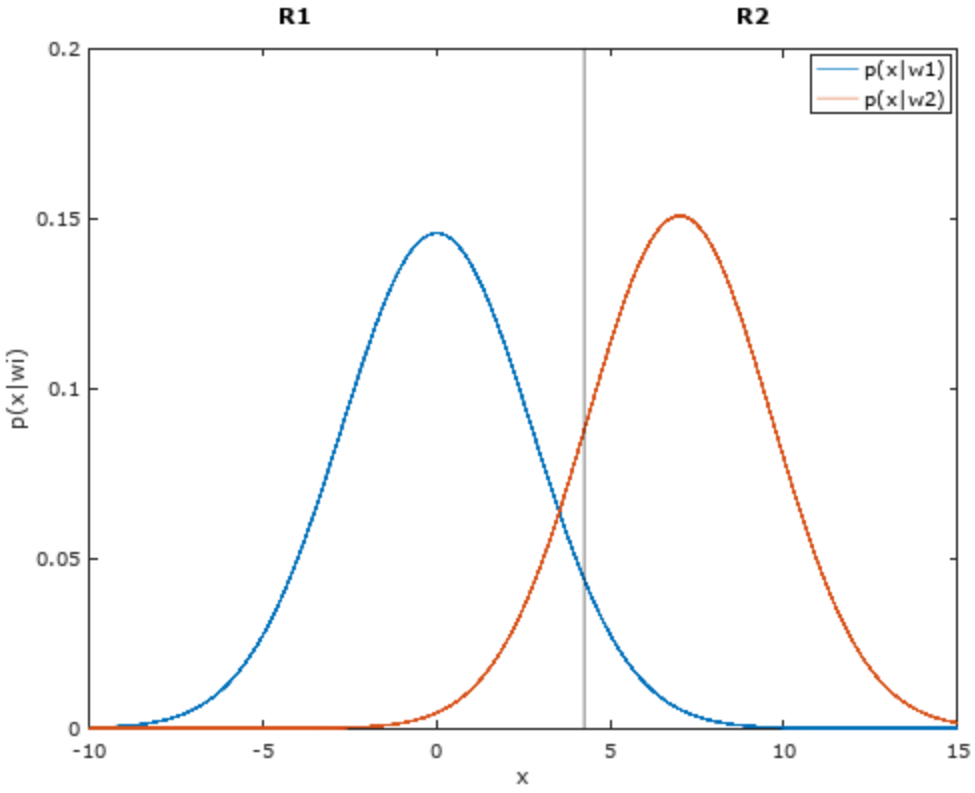
When solving for *x*, only reasonable roots will be kept.

If the boundary is at x_b, then $R_1 = \{x|x < x_b\}$ and $R_2 = \{x|x > x_b\}$.

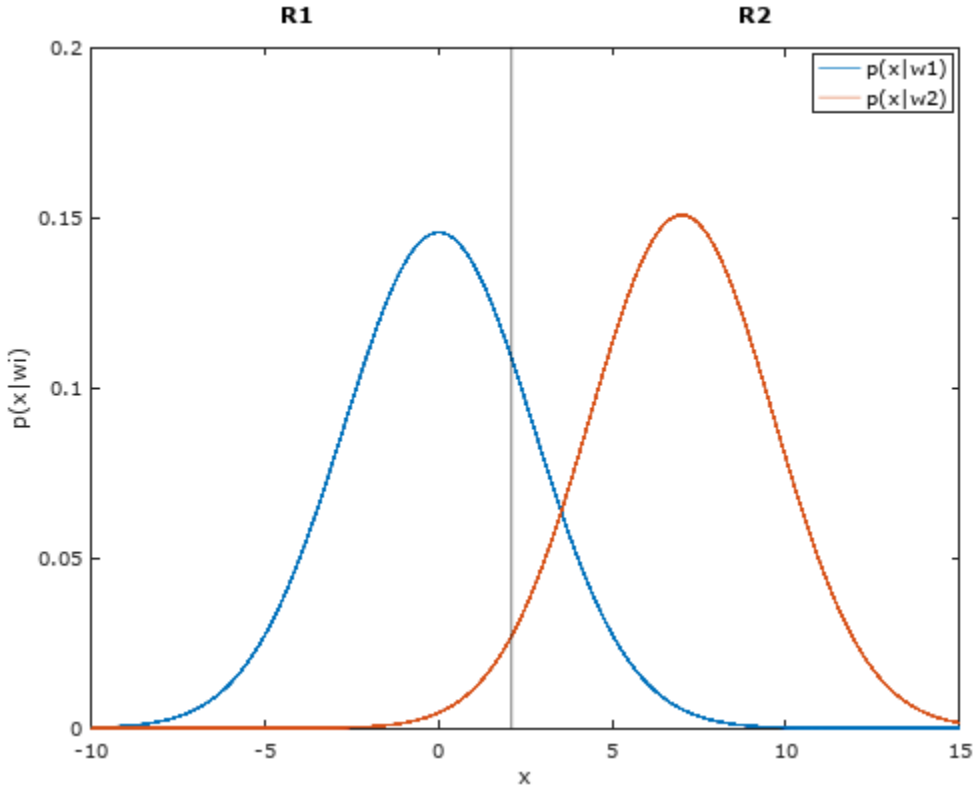
b.1) $\frac{P(w_2)}{P(w_1)} = 1 \quad \Rightarrow \quad x^2 - 210x + 727.76 = 0 \quad \Rightarrow \quad x = 3.52$



b.2) $\frac{P(w_2)}{P(w_1)} = 0.5 \quad \Rightarrow \quad x^2 - 210x + 873.32 = 0 \quad \Rightarrow \quad x = 4.24$



b.3) $\frac{P(w_2)}{P(w_1)} = 4 \implies x^2 - 210x + 436.64 = 0 \implies x = 2.1$



Problem 1.3

Let's define the following:

w_1: group of patients who are healthy

w_2: group of patients who are sick

α_1: decision that patient is healthy

α_2: decision that patient is sick

x : test result

We can find that:

$$P(w_1) = \frac{9{,}999}{10{,}000} \qquad P(w_2) = \frac{1}{10{,}000}$$

$$p(x|w_1) = \frac{1}{0.1\sqrt{2\pi}} e^{\frac{-x^2}{0.02}} \qquad p(x|w_2) = \frac{1}{0.3\sqrt{2\pi}} e^{\frac{-(x-1.5)^2}{0.18}}$$

The loss matrix is as follows:

		State of nature	
		w_1/Healthy	w_2/Sick
Decision	α_1/Healthy	$\lambda_{11} = 0$	$\lambda_{12} = 800{,}000$
	α_2/Sick	$\lambda_{21} = 1{,}500$	$\lambda_{22} = 0$

The decision rule that minimizes the conditional risk is:

$$\alpha(x) = \underset{i=1,2}{\operatorname{argmin}} R(\alpha_i|x)$$

$$\Longrightarrow R(\alpha_1|x) \underset{>w2}{\overset{<w1}{}} R(\alpha_2|x)$$

$$\Longrightarrow \frac{p(x|w_1)}{p(x|w_2)} \underset{<w2}{\overset{>w1}{}} \frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}} \frac{P(w_2)}{P(w_1)}$$

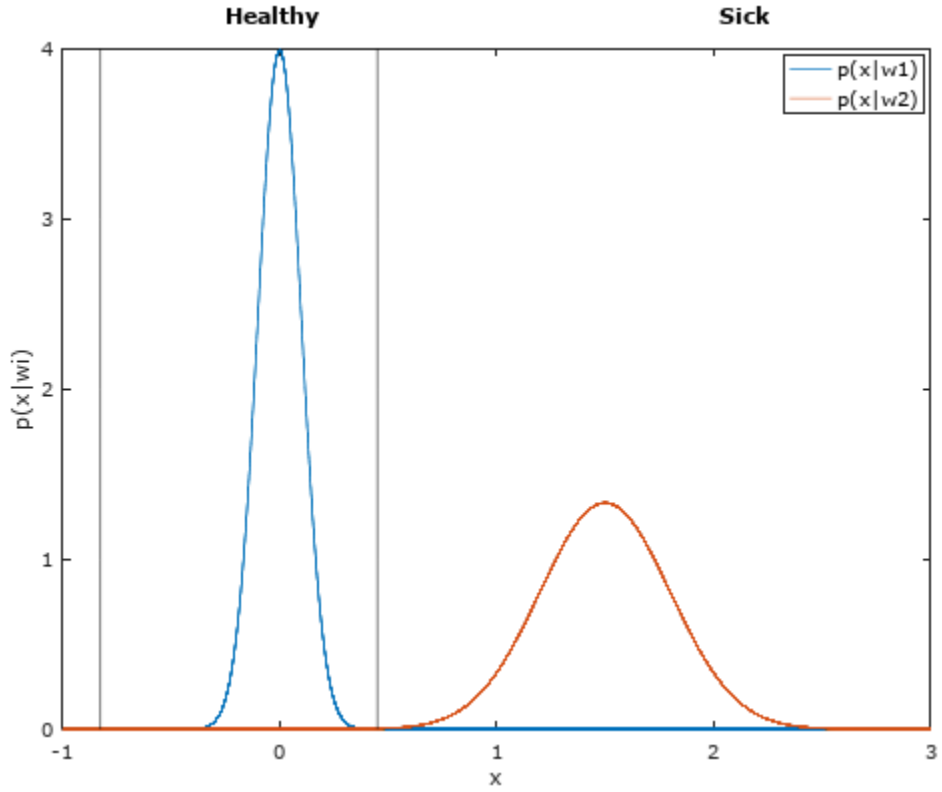
$$\Longrightarrow 3 \frac{e^{\frac{-x^2}{0.02}}}{e^{\frac{-(x-1.5)^2}{0.18}}} \underset{<w2}{\overset{>w1}{}} \frac{1}{9{,}999} \frac{800{,}000}{1{,}500}$$

$$\Longrightarrow \ln\left(\frac{e^{\frac{-x^2}{0.02}}}{e^{\frac{-(x-1.5)^2}{0.18}}}\right) \quad \gtrless_{w2}^{w1} \ln(0.018)$$

After further simplifications we get:

$$16x^2 + 6x - 5.9463 \lessgtr_{w2}^{w1} 0$$

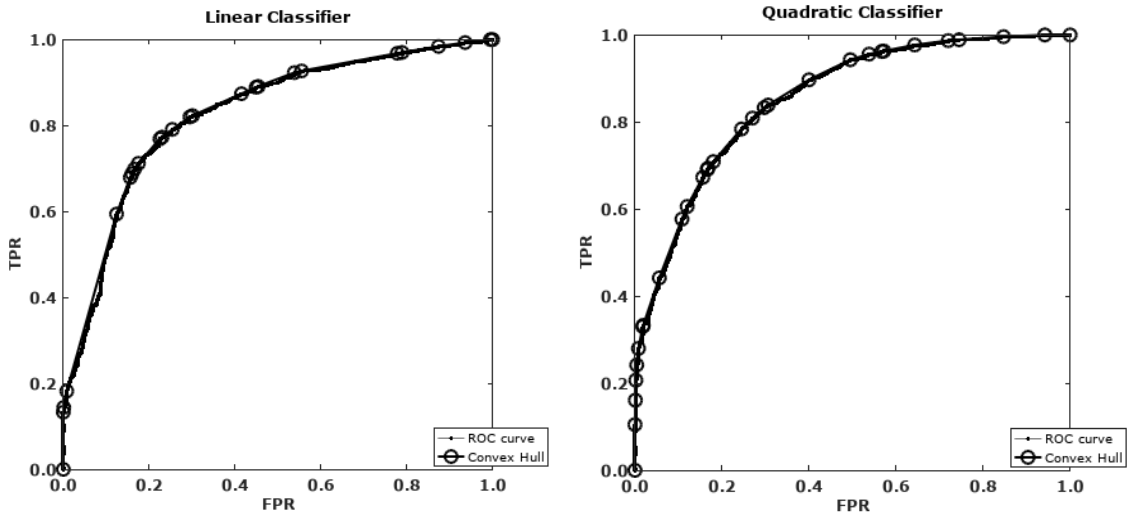
Which implies deciding for $w1$ (healthy) if $-0.825 < x\ (test\ result) < 0.450$, otherwise deciding for $w2$ (sick).



Problem 1.4

a) The averages were computed from 50 repetitions.

Linear classifier	Quadratic classifier
(FPR, TPR) = (0.1, 0.59)	(FPR, TPR) = (0.1, 0.61)
Average AUC = 0.83	Average AUC = 0.87
Average TPR = 0.55	Average TPR = 0.61
Average Accuracy = 72.7%	Average Accuracy = 71.5%



This experiment showed that the linear classifier is slightly more accurate than the quadratic, but the latter consistently provided better TPR when FPR is 0.1 and better AUC. It can be inferred that even though the linear classifier was relatively lower in sensitivity, it was very much better in specificity which lead to the higher accuracy.

Overall, while the ideal is to achieve both high sensitivity and high specificity, in reality, and through this experiment, there is a trade-off between the two such that a classifier may have the highest accuracy while not necessarily the highest TPR.

b) The averages were computed from 10 repetitions.

	K = 15	K = 25	K = 35	K = 45	K = 65	K = 75	K = 85
TPR at FPR=0.1	0.669	0.665	0.700	0.655	0.656	0.651	0.68
Average TPR	0.678	0.690	0.680	0.674	0.668	0.659	0.677
Average AUC	0.888	0.894	0.894	0.890	0.886	0.884	0.884
Average Accuracy	80.60%	80.63%	80.20%	79.79%	79.05%	78.60%	79.18%

The k value is chosen odd to avoid situations where there is an equal count from each group within the neighborhood. The experiment with k=25 yielded the highest average accuracy and average TPR (for a FPR of 0.1). It can be clearly seen that increasing the value of k is not improving the average. But overall, the results for the different k values are very close.

Moreover, the KNN with all the k values used has outperformed the linear and quadratic classifiers in terms of accuracy and TPR.

Problem 1.5

From the outcomes of the experiments in “Problem 1.4”, I decided to select the KNN classifier.

To choose a k value and train set, I have run many tests with different combination of k’s, train sets (randomly and with different sizes), and new test sets which I have generated following the distribution of the original data provided. There are different outcomes for the accuracy with each single combination which made it difficult for me to decide for which train set and k to fix for my algorithm. For example, just if the size of the test data changes with the same train set and same k value then the accuracy when the test data size is lower is getting lower unless a higher value of k is chosen which will somehow compensate.

After many trials and observations, the best I have found is with a train set of 6,800 samples (initially generated randomly from the original data with a ratio of 0.85) and a k = 29. With this I get an accuracy of at least 79% with new test sets of different sizes (e.g. 4000, 6000 and 8000 test samples).

I include a text file p5_experiment.txt which shows the outcomes of some of these experiments. I could plot the results but the graphs will be many and complicated as I did combination of 3 factors for each trial.

The function is in test.m and needs the train data in data1.mat. An implementation of calling this function is in main5.m which uses a test data of 8000 samples in testdatafile.mat.