Pattern Recognition Techniques

2013/2014

Teste Final 7 January 2014 Duration: 2h15

Name: Tiago Silva **Number:** 20222162 **Practical Class:** 19

AVISO

The Final Test has a duration of 2h15m. The test is composed by five questions. The last question is a Matlab practical question. Each question must be answered in the framed box below it. Questions may be answered in Portuguese or English. This is a closed book test. You are allowed to use a calculator machine. As consultation you may use only 1 Page A4 with your own notes. Violation of the last rule ends up with exam cancellation, course failure and eventually you may be subject to disciplinary procedure. If you have any questions, you may ask. Good Luck!

Question	pts	Results	Graded by:
1)	20		
2)	10		
3)	20		
4)	20		
5)	30		

Graded by:

Question 1 - Linear Discriminant Classifiers

□ **20 pts** Consider the (reduced) Fruits data set with 58 images, 22 apples and 36 peaches. For pattern recognition two features were extracted: the *shape ratio* that is computed by dividing the fruits width by the fruits height; and the *color ratio* that is computed by dividing the red intensity by the green intensity for a region coincident with the center of the fruits. Fig. 1 presents the distribution of the different patterns.

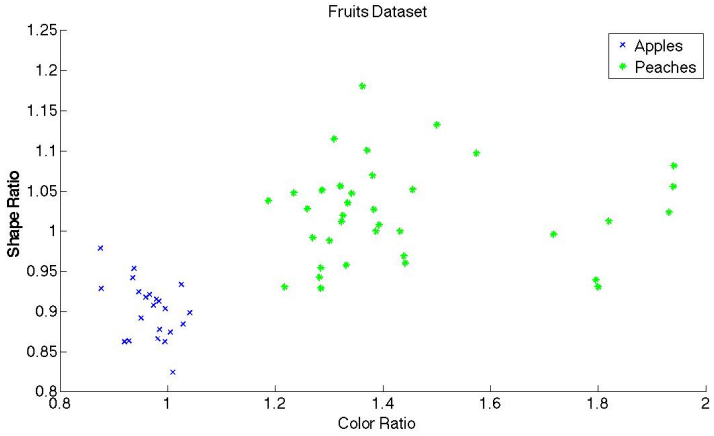


Figure 1: Fruits Dataset

1. For the maximum distance classifier write the linear decision functions using: (a) Euclidean distance; (b) Mahalanobis distance

2. Define the decisions hyperplanes for both distances. In the Euclidean case prove that the hyperplane is normal to the linear segment that connects the points defined by the two mean vectors.

Consider the following statistics for the given data:

Mean Vector	$\mathbf{m}_{\text{Apples}} = [0.9681\ 0.9021]^T$ $\mathbf{m}_{\text{Peaches}} = [1.4434\ 1.0217]^T$
Covariance matrix	$\mathbf{C}_{\text{Apples}} = \begin{bmatrix} 0.0020 & -0.0007 \\ -0.0007 & 0.0013 \end{bmatrix}$ $\mathbf{C}_{\text{Peaches}} = \begin{bmatrix} 0.0482 & 0.0004 \\ 0.0004 & 0.0036 \end{bmatrix}$
Pooled Covariance matrix	$\mathbf{C} = \begin{bmatrix} 0.0251 & -0.0002 \\ -0.0002 & 0.0025 \end{bmatrix}$
Inverse Pooled Covariance	$\mathbf{C}^{-1} = \begin{bmatrix} 39.8415 & 2.6083 \\ 2.6083 & 405.6574 \end{bmatrix}$

Your answer: Questions 1 and 2

1. Euclidean:

$$d_{1,2}(x) = (m_1 - m_2)^T [x - 0.5(m_1 + m_2)] =$$

$$= \left(\begin{bmatrix} 0.9681 \\ 0.9021 \end{bmatrix} - \begin{bmatrix} 1.4434 \\ 1.0217 \end{bmatrix} \right)^T \left[x - 0.5 \left(\begin{bmatrix} 0.9681 \\ 0.9021 \end{bmatrix} + \begin{bmatrix} 1.4434 \\ 1.0217 \end{bmatrix} \right) \right]$$

$$= [-0.4753, -0.1196]\, x - 0.5 [-0.4753, -0.1196] \begin{bmatrix} 2.4115 \\ 1.9238 \end{bmatrix}$$

$$= [-0.4753, -0.1196]\, x - 0.5 (-1,1462 - 1,3763)$$

$$= [-0.4753, -0.1196]\, x - 1,2613$$

Mahalanobis:

$$= [-19.25, -49.76]\, x - 71,1$$

$$d^M_{1,2}(x) = \underset{1\times 2}{[-0.4753, -0.1196]} \underset{2\times 2}{\begin{bmatrix} 39.846 & 2.6083 \\ 2.6083 & 405.6574 \end{bmatrix}} \left[x - 0.5 \begin{bmatrix} 2.4115 \\ 1.9238 \end{bmatrix} \right] = \underset{1\times 2}{[-19.25, -49.76]} \left[x - \underset{2\times 1}{\begin{bmatrix} 1.2067 \\ 0.962 \end{bmatrix}} \right] = \underset{1\times 2}{[-19.25, -49.76]}\, x - 71,1$$

For the Euclidean Case:
$(m_1 - m_2)$ gives the direction of the vector connecting the two means.
In the equation for the decision hyperplane: $w^T x + w_0$, w^T is the vector perpendicular to the plane.
Given that $w = (m_1 - m_2)$ we can infer the aforementioned proposition

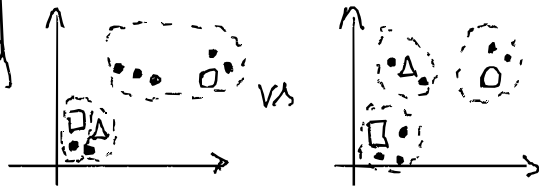
Question 2 - Unsupervised Learning

□ **10 pts** K-means clustering is very useful for many problems. Unfortunately does not always work well. (a) Explain why giving two examples justifying this issue. (b) Complement your answer - in each case - with the help of a Figure.

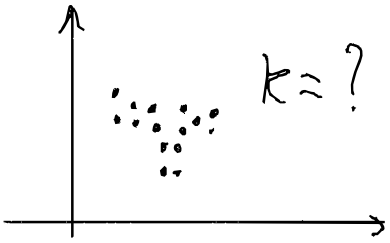
Your answer:

1. It is not deterministic (depends on the initial choice of cluster, which is random)

K=2



2. It may not be trivial to know how many clusters i want to create



Question 3 - Structural Risk Minimization

20 pts Structural Risk minimization is an inductive principle in machine learning and pattern recognition which uses the interplay of two important criteria.

1. What are these criteria? Explain this inductive principle based on these criteria. Use a schematic figure to help you on your explanation.
2. Give a formal definition of the classifier the above principle induces?

Your answer:

$\frac{FP}{FP+TN}$ FPR = .05

TPR = .85

$\frac{TP}{TP+FN}$ $P(b=1|a=1) = 0.85$

$P(b=0|a=1) = 0.15$

$P(b=1|a=0) = 0.05$

$P(b=0|a=0) = 0.95$

Question 4 - Bayes Decision Theory

□ **20 pts** You have written a face detection algorithm. Let a denote the variable that there is a face in the image and b the output of your algorithm.

$$a = \begin{cases} 1 \text{ if there is a face in the image} \\ 0 \text{ if there is not a face in the image} \end{cases} \quad b = \begin{cases} 1 \text{ if your algorithm reports there's a face in image} \\ 0 \text{ if your algorithm reports there's not a face in image} \end{cases}$$

Your face detection algorithm has a false positive rate of .05 and a true positive rate of .85. Your algorithm is examining images that are taken from your front door.

1. You run your algorithm on an image taken at 10am (the time when the postman usually passes your house) and the result is positive. What is the probability the image contains a face ?

2. You run your algorithm on an image taken at 2am and the result is positive. What is the probability the image contains a face ?

3. Write the Bayes decision rule for the minimum risk classification for the above pattern recognition image system

Your answer:

1. $P(a=1|b=1) = \frac{P(a=1) \times P(b=1|a=1)}{P(b=1)} = \frac{0.85\, P(a=1)}{P(b=1)}$

a \ b	1	0
1	TP=85	FN = 15
0	FP=5	TN = 95

Just assume a value depending on the time of the day

$P(b=1) = P(b=1, a=1) + P(b=1, a=0) =$

$= P(a=1)P(b=1|a=1) + P(a=0)P(b=1|a=0)$

→ Decide $a=1$ if: $P(a=1|b) > P(a=0|b)$

$\Rightarrow \frac{P(b|a=1)}{P(b|a=0)} > \frac{P(a=0)\,\lambda_{10}}{P(a=1)\,\lambda_{01}}$

Pattern Recognition Techniques

2013/2014

TRP Practical Question

Practical Question

□ **30 pts**

1. Consider the "FHR_APGAR.xls" dataset that contains features of foetal heart rate (FHR) tracings recorded just previous to birth of 227 newborns. The health status of the newborns was evaluated using the Apgar index, one minute (Apgar1) and five minutes (Apgar5) after birth. Lower values of Apgar mean bad prognosis. Consider that if APGAR 1 is smaller or equal than six, a bad prognosis alarm should be raised.

   Write a Matlab script by using the functionalities of the STPRTool toolbox that should:

   (a) Read the data from an Excel file named "FHR_APGAR.xls". The file has two sheets: one with the data description and named "Description", and other named "Data" which contains the values of the features of foetal heart rate and of the two Apgar indexes. Figure 1 and 2 present two pictures of the information contained in the two sheets.

   (b) Rank the features based on the area under the ROC curve (AUC).

   (c) Define a k-NN classifier that classify the patterns based on the two most discriminative features and with 8 neighbors

   (d) Split randomly the data into training and testing datasets. Consider half of the data for training and half for testing.

   (e) Assess the generalization capacity of the classifier based on statistics derived from 10 training & testing trials.

Useful Matlab Functions:

[NUM,TXT,RAW]= xlsread(FILE,SHEET)	model=knnrule(data)
I = find(X)	model=knnrule(data,K)
[FP, FN] = roc (dfce,y)	error = cerror(ypred,ytrue)
[Y,I] = sort(X,DIM,MODE)	error = cerror(ypred,ytrue,label)
y = knnclass(X,model)	P = randperm(N)

	A	B	C	D	E	F	G	H	I	J
1										
2										
3	Dataset with features of foetal heart rate (FHR) tracings recorded just previous to birth									
4	and the Apgar index evaluated just after delivery									
5										
6	All data (227 cases) has been collected in Portuguese Hospitals.									
7										
8	Hospital	HSJ - Hospital de S. João, Porto								
9		HSGA - Hospital Geral de S.to António, Porto								
10		HUC - Hospital da Universidade de Coimbra								
11	Apgar1	Apgar measured at 1 minute after (low values are bad prognostic)								
12	Apgar5	Apgar measured at 5 minutes after birth								
13										
14	Duration	Duration in minutes of the FHR tracing								
15	Baseline	Basal value of the FHR in beat/min								
16	Acelnum	Number of FHR accelerations								
17	Acelrate	Number of FHR accelerations per minute								
18	AbSTV	Percentage of the total duration with abnormal short term variability								
19	AverSTV	Average duration of the time intervals with abnormal short term variability								
20	AbLTV	Percentage of the total duration with abnormal long term variability								
21	AverLTV	Average duration of the time intervals with abnormal long term variability								
22										
23	Source:	Dr. Diogo Ayres-de-Campos, Fac. Med. Univ. Porto								
24		http://sisporto.med.up.pt/								
25										

Figure 2: Sheet 'Description"

	A	B	C	D	E	F	G	H	I	J	K	L
1	HOSPITAL	NAME	Apgar 1	Apgar 5	Duration	Baseline	Acelnum	Acelrate	AbSTV%	aver STV	abLTV%	aver LTV
2	HUC	PMGVG	9	10	44	127	3	.07	72	0.40	16	9.60
3	HUC	MCSR	8	10	55	126	23	.42	59	1.10	0	10.80
4	HUC	SMCM	9	10	46	135	9	.20	67	0.70	1	10.60
5	HUC	MFMBN	9	10	54	131	25	.46	66	0.90	0	8.70
6	HUC	EMSO	9	10	47	142	12	.26	61	0.80	10	7.60
7	HUC	CMMRJ	9	10	47	130	11	.23	68	0.80	16	8.60
8	HUC	CMRAM	9	10	39	131	6	.15	61	0.90	0	12.60
⋮	⋮					⋮					⋮	
222	HSJ	PCC	3	6	43	171	0	.00	77	0.50	54	5.10
223	HSJ	SCB	5	9	40	141	1	.03	78	0.60	6	11.50
224	HSJ	MFC	6	9	50	131	6	.12	61	2.10	2	20.50
225	HSJ	MMB	2	7	43	135	0	.00	85	0.60	75	5.00
226	HSJ	AACST	4	7	52	125	0	.00	86	0.40	38	8.00
227	HSJ	JMP	1	5	40	123	0	.00	88	0.30	71	5.70
228	HSJ	AMRC	5	8	60	141	0	.00	75	1.10	12	14.30

Figure 3: Sheet "Data"

Your answer:

```
df = pd.read_excel('FHR_APGAR.xls', sheet_name='Data')
target = (df['Apgar 1'] <= 6).astype(int)
features = [...]  X = df[features]
auc_scores = {}
for col in features:
    score = roc(y, X[col])
    auc_scores[col] = max(score, 1-score)
ranked_auc = sorted(auc_scores.items, key = lambda x: x[1], reverse =TRUE)
knn = KNC(8)                for _ in range(10):
X_train, X_test, y_train, y_test = train_test_split(...)
knn.fit(X_train, y_train)        ŷ = knn.predict(X_test)
knn.score(X_test, y_test) -> store in ar
f1_list.append(f1(target, ŷ))
np.mean(ar)
np.std(ar)
```