PRAC2: LECTURA DE BASES DE DATOS

Asignatura: CLP: Clasificación de Patrones.
Optativa de 2º Ciclo
ETSETB
UPC

Profesores:
Margarita Cabrera
Josep Vidal
UPC-TSC-D5

Febrero-2006

1 Objetivos de la práctica 2

Se observan algunas de las bases de datos con que se va a trabajar en las clases de laboratorios.
Las bases de datos se han obtenido a través del siguiente link:

http://www-stat.stanford.edu/~tibs/ElemStatLearn/data.html

Se propone reducir la dimensión de las bases de datos atendiendo a criterios PCA.

2 Base de datos: Phoneme

2.1 Características de la base de datos

Esta es la información que los autores de la base de datos facilitan sobre la misma:

```
These data arose from a collaboration between  Andreas Buja, Werner
Stuetzle and Martin Maechler, and we used as an illustration in the
paper on Penalized Discriminant Analysis by Hastie, Buja and
Tibshirani (1995), referenced in the text.

The data were extracted from the TIMIT database (TIMIT
Acoustic-Phonetic Continuous Speech Corpus, NTIS, US Dept of Commerce)
which is a widely used resource for research in speech recognition.  A
dataset was formed by selecting five phonemes for
classification based on digitized speech from this database.  The
phonemes are transcribed as follows: "sh" as in "she", "dcl" as in
"dark", "iy" as the vowel in "she", "aa" as the vowel in "dark", and
"ao" as the first vowel in "water".  From continuous speech of 50 male
speakers, 4509 speech frames of 32 msec duration were selected,
approximately 2 examples of each phoneme from each speaker.  Each
speech frame is represented by 512 samples at a 16kHz sampling rate,
and each frame represents one of the above five phonemes.  The
breakdown of the 4509 speech frames into phoneme frequencies is as
follows:

  aa   ao dcl   iy  sh
 695 1022 757 1163 872

From each speech frame, we computed a log-periodogram, which is one of
several widely used methods for casting speech data in a form suitable
for speech recognition.  Thus the data used in what follows consist of
4509 log-periodograms of length 256, with known class (phoneme)
memberships.

The data contain 256 columns labelled "x.1" - "x.256", a response
column labelled "g", and a column labelled "speaker" identifying the
different speakers.
```

Para su tratamiento se ha elegido un subconjunto de 100 espectros grabados para cada uno de los 5 fonemas diferentes.
El vector obtenido se ha recortado a los primeros 128 puntos, lo que supone una frecuencia máxima de 4 KHz.
El resultado de elegir este subconjunto se ha guardado en el fichero de texto: "phoneme_100".
En los dos ficheros de Matlab, que se comentan a continuación la base de datos se divide a su vez en dos bases:

Base de Datos de TRAIN: 250 Vectores
Base de Datos de TEST: 250 Vectores.

1 El fichero de matlab: "prac2_fonemas", realiza la lectura de estos datos, presenta el dibujo de los diferentes espectros y evalúa algún otro parámetro de clasificación. Subrutina de clasificación: ldc. La división entre base de TRAIN y base de TEST, se realiza de forma determinista.
2 El fichero de matlab: "prac2_fonV", es equivalente al anterior pero trabajando únicamente con 2 fonemas a elegir. Subrutina de clasificación: ldc, qdc. La división entre base de TRAIN y base de TEST, de forma aleatoria.

2.2 Laboratorio

PARTE 1
A realizar en el laboratorio mediante el programa "prac2_fonemas":

- Observe los diferentes espectros obtenidos para los diferentes fonemas.
- Observe el scatter de algunas coordenadas elegidas mediante el vector "V_coor". Comente si solo pudiera trabajar con 2 coordenadas de cada vector cuales elegiría. Para buscar 2 coordenadas muy discriminativas puede ayudarse tanto del "scatter" plot como de las gráficas de espectro.
- Observe los errores de clasificación al aplicar ldc.
- Compare la mayor y la menor distancia de Mahalanobis entre clases y elija 1) El par de clases mas distanciadas entre si. 2) El par de clases menos distanciadas entre si.

PARTE 2

A realizar en el laboratorio mediante el programa "prac2_fonV":
Mediante este programa al inicio del mismo se seleccionan clases y coordenadas. Por ello lo puede ejecutar eligiendo

- Las dos clases de mayor distancia entre sí
- Las dos clases de menor distancia entre sí

Además puede elegir cualquier vector de coordenadas mediante V_coor y los algoritmos de clasificación se ejecutarán eligiendo únicamente las coordenadas del vector V_coor. Es

interesante probar el caso de las dos coordenadas que crea más discriminativas a partir de los diagramas espectrales.

Para cada caso evaluar:

- Error de Clasificación al aplicar ldc y qdc para la base de TRAIN y para la base de TEST.
- ROC al aplicar ldc y qdc para la base de TRAIN y para la base de TEST.
- Scatter al trabajar con 2 coordenadas y fronteras de clasificación, al aplicar ldc y qdc para la base de TRAIN y para la base de TEST.
- Autovalores obtenidos para las matrices de covarianza de cada una de las dos clases (Para ver dichas matrices active la opción "anaclass=1").

Autoval=eig(squeeze(Cov(:,:,i))) %Estimated

COMENTE LOS RESULTADOS EN EL DOCUMENTO: G***_prac2.doc

3 Base de Datos SPAM

Esta base de datos se halla formada por 4601 vectores obtenidos de 4061 e-mails. A partir de la frecuencia con que aparece cada palabra se debe predecir si el e-mail es SPAM o no lo es. Cada vector es de 57 coordenadas, gran dimensionalidad, por lo que interesa identificar las características más significativas.

3.1 Características de la base de datos

Esta es la información que los autores de la base de datos facilitan sobre la misma:

```
1. Title:  SPAM E-mail Database

2. Sources:
  (a) Creators: Mark Hopkins, Erik Reeber, George Forman, Jaap Suermondt
      Hewlett-Packard Labs, 1501 Page Mill Rd., Palo Alto, CA 94304
  (b) Donor: George Forman (gforman at nospam hpl.hp.com)  650-857-7835
  (c) Generated: June-July 1999

3. Past Usage:
  (a) Hewlett-Packard Internal-only Technical Report. External forthcoming.
  (b) Determine whether a given email is spam or not.
  (c) ~7% misclassification error.
     False positives (marking good mail as spam) are very undesirable.
     If we insist on zero false positives in the training/testing set,
     20-25% of the spam passed through the filter.

4. Relevant Information:
"       The ""spam"" concept is diverse: advertisements for products/web"
       sites, make money fast schemes, chain letters, pornography...
            Our collection of spam e-mails came from our postmaster and
            individuals who had filed spam.  Our collection of non-spam
            e-mails came from filed work and personal e-mails, and hence
            the word 'george' and the area code '650' are indicators of
            non-spam.  These are useful when constructing a personalized
            spam filter.  One would either have to blind such non-spam
            indicators or get a very wide collection of non-spam to
            generate a general purpose spam filter.

       For background on spam:
       Cranor, Lorrie F., LaMacchia, Brian A.  Spam!
       Communications of the ACM, 41(8):74-83, 1998.

5. Number of Instances: 4601 (1813 Spam = 39.4%)

6. Number of Attributes: 58 (57 continuous, 1 nominal class label)

7. Attribute Information:
The last column of 'spambase.data' denotes whether the e-mail was
considered spam (1) or not (0), i.e. unsolicited commercial e-mail.
Most of the attributes indicate whether a particular word or
character was frequently occuring in the e-mail.  The run-length
attributes (55-57) measure the length of sequences of consecutive
```

capital letters. For the statistical measures of each attribute,
see the end of this file. Here are the definitions of the attributes:

48 continuous real [0,100] attributes of type word_freq_WORD
= percentage of words in the e-mail that match WORD,
i.e. 100 * (number of times the WORD appears in the e-mail) /
"total number of words in e-mail. A ""word"" in this case is any "
string of alphanumeric characters bounded by non-alphanumeric
characters or end-of-string.

6 continuous real [0,100] attributes of type char_freq_CHAR
= percentage of characters in the e-mail that match CHAR,
i.e. 100 * (number of CHAR occurences) / total characters in e-mail

1 continuous real [1,...] attribute of type capital_run_length_average
#¿NOMBRE?

1 continuous integer [1,...] attribute of type capital_run_length_longest
#¿NOMBRE?

1 continuous integer [1,...] attribute of type capital_run_length_total
#¿NOMBRE?
#¿NOMBRE?

1 nominal {0,1} class attribute of type spam
= denotes whether the e-mail was considered spam (1) or not (0),
i.e. unsolicited commercial e-mail.

8. Missing Attribute Values: None

9. Class Distribution:
Spam 1813 (39.4%)
Non-Spam 2788 (60.6%)

NEXT TABLE shows the content of feature vector (dimension = 57)

This file: 'spambase.DOCUMENTATION' at the UCI Machine Learning Repository
http://www.ics.uci.edu/~mlearn/MLRepository.html

Number	Feature	Number	Feature
1	word_freq_make: continuous.	30	word_freq_labs: continuous.
2	word_freq_address: continuous.	31	word_freq_telnet: continuous.
3	word_freq_all: continuous.	32	word_freq_857: continuous.
4	word_freq_3d: continuous.	33	word_freq_data: continuous.
5	word_freq_our: continuous.	34	word_freq_415: continuous.
6	word_freq_over: continuous.	35	word_freq_85: continuous.
7	word_freq_remove: continuous.	36	word_freq_technology: continuous.
8	word_freq_internet: continuous.	37	word_freq_1999: continuous.
9	word_freq_order: continuous.	38	word_freq_parts: continuous.
10	word_freq_mail: continuous.	39	word_freq_pm: continuous.
11	word_freq_receive: continuous.	40	word_freq_direct: continuous.
12	word_freq_will: continuous.	41	word_freq_cs: continuous.
13	word_freq_people: continuous.	42	word_freq_meeting: continuous.
14	word_freq_report: continuous.	43	word_freq_original: continuous.
15	word_freq_addresses: continuous.	44	word_freq_project: continuous.
16	word_freq_free: continuous.	45	word_freq_re: continuous.
17	word_freq_business: continuous.	46	word_freq_edu: continuous.
18	word_freq_email: continuous.	47	word_freq_table: continuous.
19	word_freq_you: continuous.	48	word_freq_conference: continuous.
20	word_freq_credit: continuous.	49	char_freq_;: continuous.
21	word_freq_your: continuous.	50	char_freq_(: continuous.
22	word_freq_font: continuous.	51	char_freq_[: continuous.
23	word_freq_000: continuous.	52	char_freq_!: continuous.
24	word_freq_money: continuous.	53	char_freq_$: continuous.
25	word_freq_hp: continuous.	54	char_freq_#: continuous.
26	word_freq_hpl: continuous.	55	capital_run_length_average: continuous.
27	word_freq_george: continuous.	56	capital_run_length_longest: continuous.
28	word_freq_650: continuous.	57	capital_run_length_total: continuous.
29	word_freq_lab: continuous.		

3.2 Laboratorio

A realizar en el laboratorio:

- Ejecute “prac2_spam_pca”.
- Intente seleccionar un sub-conjunto de características que mejore los errores de clasificación, ya sea a partir del dibujo de las mismas o a partir de la información facilitada en clase sobre esta base de datos y del vector “V_coor”.
- Observe los errores de clasificación al aplicar ldc y qdc, con el vector de características entero y con el seleccionado.
- Evalúe la ROC con el vector de características entero y con el seleccionado.

COMENTE LOS RESULTADOS EN EL DOCUMENTO: G***_prac2.doc. Si en algún caso, el criterio qdc funciona pero que el criterio ldc, calcule *cond* de las correspondientes matrices de covarianza para analizar como se hallan condicionadas estas matrices.

3. Reducción de Dimensión mediante PCA

Observe que las dos bases de datos utilizadas en esta práctica, son de dimensión grande (64 y 57).
En las dos partes anteriores se ha reducido la dimensión, en función de la observación directa de las coordenadas. En esta parte se propone reducir la dimensión justificadamente, atendiendo a un criterio de componentes principales. Utilice para ello la subrutina *pca.m*. En el fichero *prac2_spam_pca.m*, se muestra un ejemplo de llamada a esta subrutina.

Presente para la base de datos SPAM (d=57):

- Una gráfica del error obtenido en clasificación con ldc para la base de datos de train al variar la dimensión de 1:d;
- Una gráfica del error obtenido en clasificación con ldc para la base de datos de test al variar la dimensión de 1:d;
- Una gráfica del error obtenido en clasificación con qdc para la base de datos de train al variar la dimensión de 1:d;
- Una gráfica del error obtenido en clasificación con qdc para la base de datos de test al variar la dimensión de 1:d;

Programe de tal modo que los cuatro errores pedidos aparezcan en la misma gráfica.

Presente para la base de datos FONEMAS (d=64):

- Una gráfica del error obtenido en clasificación con ldc para la base de datos de train al variar la dimensión de 1:d;
- Una gráfica del error obtenido en clasificación con ldc para la base de datos de test al variar la dimensión de 1:d;
- Una gráfica del error obtenido en clasificación con qdc para la base de datos de train al variar la dimensión de 1:d;
- Una gráfica del error obtenido en clasificación con qdc para la base de datos de test al variar la dimensión de 1:d;

Programe de tal modo que los cuatro errores pedidos aparezcan en la misma gráfica.
COMENTE LOS RESULTADOS EN: G***_prac2.doc (Entregue un único WORD para las 3 partes).

Opcional:
Como alternativa, se pueden aplicar otras técnicas sub-óptimas de reducción de características. Realice algún caso particular de "featselb.m" o "feaself.m", correspondientes a los criterios de forward y backward selection.