

Instituto tecnológico de Massachusetts
Departamento de ingeniería eléctrica e informática

6.345 Reconocimiento automático del habla
Primavera 2003

Publicado: 28/02/03
Entregar: 12/03/03

Trabajo 4
Fonética acústica, modelado acústico y
clasificación de patrones

Introducción

El propósito de este trabajo es afianzar sus conocimientos de fonética acústica, modelado acústico y clasificación de patrones. Investigará primero las propiedades acústicas de los sonidos fricativos del inglés [s z š ž f v θ ð] , examinando las mediciones acústicas tomadas de ejemplos de estos fonos, extraídos del corpus TIMIT. Luego, investigará varias técnicas distintas para modelar estas mediciones, y llevará a cabo un conjunto de experimentos de clasificación de patrones, utilizando estas técnicas.

Al igual que en trabajos anteriores, las siguientes tareas (marcadas con **T**) deberían acabarse durante la sesión de prácticas. Las respuestas a las preguntas (señaladas con **P**) deberían entregarse dentro del plazo estipulado.

Cómo comenzar

El trabajo 4 se llevará a cabo utilizando MATLAB®, un paquete de software matemático creado por MathWorks, Inc. Esta fotocopia contiene suficiente documentación para completar los ejercicios de prácticas. La información adicional está disponible en línea mediante el comando `help` dentro de MATLAB®. Para comenzar la práctica, escriba lo siguiente en la línea de comandos de UNIX:

```
% start_lab4.cmd
```

Con este comando arrancará MATLAB®. Después de la inicialización, aparecerá la línea de comandos de MATLAB. En esta línea, escriba el siguiente comando:

```
>> init_lab4
```

Este comando cargará los datos que usted utilizará durante el trabajo.

Descripción de los datos

Los datos utilizados en este trabajo fueron extraídos del corpus TIMIT. Para el análisis de este trabajo, se seleccionó un grupo de ocho fonos distintos, concretamente las fricativas [s z š ž f v θ ð], . En MATLAB®, las etiquetas 's', 'z', 'sh', 'zh', 'f', 'v', 'th y 'dh' respectivamente, se utilizaban para representar a estos ocho fonos. Se creó un vector de observación de seis mediciones para cada ejemplo de estos fonos en el corpus. Estas seis mediciones son:

- **MFCC0**: El valor medio del coeficiente MFCC cero.
- **MFCC1**: El valor medio del primer coeficiente MFCC .
- **TOT Energy**: La energía logarítmica total media.
- **LF Energy**: La energía logarítmica media entre 50 Hz y 200 Hz (energía de baja frecuencia).
- **MFR Energy**: La proporción de energía logarítmica entre 2kHz y 4kHz y entre 2 kHz y 6 kHz (energía de proporción de frecuencia media).
- **Duración**: La duración logarítmica del fono.

Se han creado distintas agrupaciones de los fonos mostrados para facilitar el análisis durante la tarea. Después de que los datos del trabajo 4 se hayan cargado, escriba el siguiente comando:

```
>> whos
```

Este comando enumerará los nombres de variables de MATLAB para cada porción de datos utilizados en el trabajo. Lo que se muestra a continuación es una descripción de las variables:

- Para la parte I del trabajo:
 - **fric_AP**: Un conjunto de vectores para los fonos [s z š ž] (donde AP hace referencia a la alveolar y palatal). Cada fila de la matriz corresponde a una instancia del fono y las columnas contienen las mediciones de rasgos para cada instancia.
 - **fric_AP_index**: Un conjunto de índices que indican la “clase de fono” con la que cada vector fila en **fric_AP** está asociado. Estas etiquetas dividen los datos en dos clases: alveolar y palatal. Esta matriz también contiene información sobre los fonos de contexto izquierdo y derecho.
 - **fric_SW**: Un conjunto de vectores para los fonos [s š f θ] (SW = fuerte/débil).
 - **fric_SWindex**: Un conjunto de índices que indican la “clase de fono” con la que cada vector fila en **fric_SW** está asociado. Divide los datos en dos clases: fuerte y débil.
 - **fric_VU**: Un conjunto de vectores para los fonos [z v s f] (VU = sonoro/sordo).
 - **fric_VUindex**: Un conjunto de índices que indican la “clase de fono” con la que cada vector fila en **fric_VU** está asociado. Divide los datos en dos clases: sonoro y sordo.
- Para las partes II y III del trabajo:
 - **train**: El grupo de vectores de entrenamiento para los fonos [zš f ð]. Cada fila de la matriz es una instancia del fono y las columnas contienen las mediciones de rasgos.

- **train_index**: Un conjunto de índices que indican la "clase de fono" con la que cada vector fila en **train** está asociado. Contiene también información sobre los fonos del contexto izquierda y derecha.
- **test**: El conjunto de vectores de prueba para los fonos [z š f ð].
- **test_index**: Un conjunto de índices que indican la "clase de fonos" con la que cada vector fila en **test** está asociado.
- **train_<label>**: Cada grupo de esta forma contiene sólo los vectores de entrenamiento para un fono concreto, ej., **train_z** contiene los vectores de entrenamiento para el fono [z].
- **train_<label>_index**: Un conjunto de índices que indican la "clase de fono" con la que cada vector fila en **train_<label>** está asociado.
- **train_<N>**: Cada grupo de esta forma contiene un subconjunto de vectores de entrenamiento. $N = 1$ posee 77 vectores, $N = 2$ posee 152 vectores, $N = 3$ posee 302 vectores y $N = 4$ posee 601 vectores.
- **train_<N>_index**: Un conjunto de índices que indican la "clase de fono" con la que cada vector fila en **train_<N>** está asociado.

- Variables adicionales:

- **measurements**: Un vector de cadenas que contiene una etiqueta para cada medición de rasgos. Esto lo utilizan internamente algunas rutinas de trazado para etiquetar los ejes.
- **phone_labels**: Un vector de cadenas que contiene una etiqueta para cada índice de fono. Esto se pasa a algunas rutinas de trazado si lo que se desea es etiquetar los puntos de datos individuales.
- **phone_colors**: Un vector de cadenas que indica el color y el tipo de símbolo a utilizar para trazar cada fono listado en **phonelabels**. Esto lo utilizan internamente algunas rutinas de trazado para colorear puntos y texto.

Para observar los contenidos de cualquiera de estas variables, introduzca simplemente el nombre de la variable en la línea de comandos de MATLAB. Por ejemplo, para las **mediciones** de la variable:

```
>> measurements (mediciones)

measurements =

MFCC0
MFCC1
TOT Energy (energía total)
LF Energy (energía de baja frecuencia)
MFR Energy (energía de frecuencia media)
Duration (duración)
```

Dado que muchas de las variables contienen enormes cantidades de datos, es posible que desee observar sólo pequeñas porciones de todos los datos de una vez. Puede hacer esto expresando un rango para tener en cuenta. Por ejemplo, para la variable **train**, podemos considerar los tres primeros vectores del grupo con el comando:

```
>> train(1:3,:)
```

```
ans =
```

```
8.8863 -1.6315 7.7928 5.5401 17.6856 8.0378
7.8264 -1.2994 7.3976 4.8848 16.4082 8.1038
9.0345 -2.1741 8.4573 5.8948 16.9312 7.3185
```

Lo siguiente es una tabla de los símbolos IPA para los fonos y las correspondientes etiquetas de TIMIT, junto con ejemplos de apariciones. Esto se facilita como referencia para que usted pueda comprender lo que significan las etiquetas de fonos TIMIT cuando le aparecen en las líneas de los trabajos.

IPA	TIMIT	Ejemplo	IPA	TIMIT	Ejemplo
[ɑ]	aa	<i>bob</i>	[ɪ]	ix	<i>debit</i>
[æ]	ae	<i>bat</i>	[i]	iy	<i>beet</i>
[ʌ]	ah	<i>but</i>	[j]	jh	<i>joke</i>
[ɔ]	ao	<i>bought</i>	[k]	k	<i>key</i>
[ɑ ^w]	aw	<i>bout</i>	[k ^ɹ]	kcl	k closure
[ə]	ax	<i>about</i>	[l]	l	<i>lay</i>
[ə ^h]	ax-h	<i>potato</i>	[m]	m	<i>mom</i>
[ə ^r]	axr	<i>butter</i>	[n]	n	<i>noon</i>
[ɑ ^r]	ay	<i>bite</i>	[ŋ]	ng	<i>sing</i>
[b]	b	<i>bee</i>	[r]	rx	<i>winner</i>
[b ^ɹ]	bcl	b closure	[o]	ow	<i>boat</i>
[ç]	ch	<i>choke</i>	[ɔ ^r]	oy	<i>boy</i>
[d]	d	<i>day</i>	[p]	p	<i>pea</i>
[d ^ɹ]	dcl	d closure	[ɔ]	pau	<i>pause</i>
[ð]	dh	<i>then</i>	[p ^ɹ]	pcl	p closure
[r]	dx	<i>muddy</i>	[ʔ]	q	glottal stop
[ɛ]	eh	<i>bet</i>	[r]	r	<i>ray</i>
[l]	el	<i>bottle</i>	[s]	s	<i>sea</i>
[m]	em	<i>bottom</i>	[ʃ]	sh	<i>she</i>
[n]	en	<i>button</i>	[t]	t	<i>tea</i>
[ŋ]	eng	<i>Washington</i>	[t ^ɹ]	tcl	t closure
[ʌ]	epi	epenthetic silence	[θ]	th	<i>thin</i>
[ə ^r]	er	<i>bird</i>	[ɔ]	uh	<i>book</i>
[e]	ey	<i>bait</i>	[u]	uw	<i>boot</i>
[f]	f	<i>fin</i>	[ü]	ux	<i>toot</i>
[g]	g	<i>gay</i>	[v]	v	<i>van</i>
[g ^ɹ]	gcl	g closure	[w]	w	<i>way</i>
[h]	hh	<i>hay</i>	[y]	y	<i>yacht</i>
[h̃]	hv	<i>ahead</i>	[z]	z	<i>zone</i>
[ɪ]	ih	<i>bit</i>	[z̃]	zh	<i>azure</i>
-	h#	utterance initial and final silence			

Funciones y comandos

Para examinar y manipular los datos, se ha creado un grupo de comandos útiles para el modelado acústico y la clasificación de patrones. Para obtener una lista de estos comandos escriba:

```
>> help assign4
```

Para leer la descripción de una página del manual de cualquiera de estos comandos, escriba:

```
>> help <command_name>
```

Guardando trazados como archivos postscript (.ps)

Puede guardar el trazado actual mostrado en la ventana de ilustraciones de MATLAB®, utilizando el comando:

```
>> print -deps matlab_plot.eps
```

Esto guardará el trazado actual en formato postscript encapsulado en el archivo `matlab_plot.eps`. Para más información sobre el comando `print`, escriba `help print` en la línea de comandos de MATLAB®.

Ventanas de trazados múltiples

Por defecto, MATLAB® utiliza una “ventana de ilustración” para el trazado de líneas. Puede crear ventanas de ilustración adicionales para mostrar múltiples trazados simultáneamente mediante el comando `figure`. Escriba `help figure` en la línea de comandos de MATLAB para más información sobre cómo utilizar este comando.

Si no, si usted desea generar múltiples “subtrazados” dentro de una ventana de ilustración, puede utilizar el comando `subplot`. Escriba `help subplot` en la línea de comandos de MATLAB® para obtener más información sobre el uso de este comando.

Parte I: Fonética acústica

Comprender las propiedades de las mediciones acústicas y su relación con los distintos fonos es un primer paso importante para determinar cómo modelar los rasgos/mediciones y qué enfoque debería utilizarse para la clasificación de patrones.

En esta parte del trabajo, usted investigará las propiedades acústicas de los sonidos fricativos del inglés [s z š ž f v θ ð]. Examinaremos:

- cómo los contrastes fonéticos pueden estar relacionados con varias mediciones acústicas, y
- las influencias del contexto en las propiedades acústicas de los sonidos fricativos del discurso.

Antes de comenzar con esta parte del trabajo, le recomendamos que vuelva a familiarizarse con las propiedades generales fonético-acústicas de las fricativas. Por favor, consulte las clases sobre fonética acústica.

Fricativas fuertes frente a fricativas débiles

Comenzaremos nuestra investigación examinando los correlatos acústicos del rasgo *estriidente*. Las estridentes o fricativas fuertes son [s z š ž], y las no estridentes o fricativas débiles son [f v θ ð]. Para esta tarea, utilizaremos el grupo de datos **fric_SW**, que consta de fricativas [s š f θ] divididas en dos clases: fuertes y débiles.

T1: Genere trazados de histograma (mediante `plot_histograms`) de este grupo de datos para cada uno de las seis mediciones distintas y examine su capacidad de separar las dos clases. Por ejemplo, para crear un trazado de histograma mediante la primera medición de rasgos (**MFCC0**), utilice el siguiente comando:

```
>> plot_histograms(fric_SW,1,fric_SW_index);
```

Para la medición de la **energía total media**, los histogramas superpuestos para las dos clases deberían revelar distintas distribuciones considerables para las fricativas fuertes frente a las débiles. Sin embargo, debería existir algún solapamiento entre ellas. Miraremos ahora más de cerca a algunas de las fricativas *débiles* fuertes poco comunes, cuya **energía total media** las sitúa en la región de solapamiento. Esto se logra mejor utilizando trazados dispersos.

T2: Genere un trazado disperso de dos dimensiones con el comando `plot_clusters` para los datos **fric_SW**. Por ejemplo, para generar un trazado disperso de las dos primeras dimensiones de los datos, utilice el siguiente comando:

```
>> plot_clusters(fric_SW,1,2,fric_SW_index);
```

Los trazados dispersos le permiten observar cómo las mediciones múltiples interactúan entre sí. En este trazado, cada manifestación del fono está representada por un símbolo coloreado en las coordenadas apropiadas. Explore distintos pares de mediciones y examine sus habilidades para discriminar entre las dos clases. Tenga en cuenta cómo el uso combinado de dos mediciones puede conducir a una mejor separación de las dos clases, que si se utiliza una única medición simple.

T3: Vuelva a trazar la línea dispersa, utilizando las dos mediciones más exigentes derivadas de **T2** como dimensiones del trazado. Una de ellas debería ser la **energía total media**. Esta vez, trace también las etiquetas del fono de los puntos de datos incluyendo `phone_labels` en la llamada a `plot_clusters` como se muestra (esta línea puede generarse algo más lentamente):

```
>> plot_clusters(fric_SW,2,3,fric_SW_index,phone_labels);
```

Las etiquetas trazadas incluyen información no sólo sobre la identidad del fono "medio" especificado, sino también sobre la identidad de sus vecinos izquierdos y derechos. Las etiquetas tienen el siguiente formato: `leftmiddle_right`.

Enfoque (haciendo clic con el botón izquierdo del ratón sobre la región de interés de la figura; haga clic con el botón medio para alejar el zoom) la región de solapamiento donde las fricativas débiles tienen una gran **energía total media**. Es posible que quiera volver a modificar el tamaño de la línea para obtener una mejor visión. Estudie los contenidos fonéticos de estas fricativas débiles.

P1: ¿Puede identificar un entorno fonético en el que aparezcan las fricativas fuertes [f θ] poco comunes? Para esta y todas las cuestiones posteriores similares, si tiene problemas al determinar los contextos pertinentes desde el único trazado disperso, trate de plantear algunas ideas basadas en lo que ha visto en clase sobre fonética acústica, y pruebe sus hipótesis observando el trazado disperso. Si hay cualquier efecto contextual que usted crea que debiese estar allí, pero que no se manifiesta en el trazado disperso, trate de describir los resultados y por qué puede que no sea evidente.

Fricativas sordas frente a sonoras

Seguiremos un procedimiento parecido para contrastar las distribuciones de las fricativas *sonoras* [zʒvð] y sordas [s š f θ]. Para esta tarea, utilizaremos el grupo de datos **fric_VU**, que consta de las fricativas [z v s f] divididas en dos clases: sonoras y sordas.

T4: Utilizando el grupo de datos **fric_VU**, examine los trazados del histograma (utilizando `plot_histograms`) para cada una de las seis mediciones, y estudie su habilidad para separar las dos clases.

Dése cuenta que mientras que las distribuciones de **duración** para las fricativas sordas y sonoras se distinguen, no existe solapamiento sustancial entre ellas. Esto se debe al hecho de que las propiedades acústicas de los fonemas se ven afectadas por los entornos en los que éstos aparecen.

T5: Genere un trazado disperso de dos dimensiones utilizando el comando `plot_clusters` para los datos **fric_VU**, utilizando las mediciones de **duración** y **energía de baja frecuencia**. Incluya las etiquetas de los fonos en el trazado.

P2: A partir del conjunto de contextos disponibles facilitados en la muestra de datos, trate de definir dos factores que tengan una influencia significativa en las duraciones de las fricativas, ej., ¿qué contextos pueden provocar el alargamiento de la fricativa? ¿Y el acortamiento?

T6: Las fricativas sordas y sonoras también se distinguen por la cantidad de flujo defrecuencia energía. Estudie esta diferencia acústica utilizando el mismo procedimiento señalado arriba, mediante la medición de energía de 50 a 200 Hz de **energía de baja frecuencia**.

P3: ¿Existen entornos fonéticos que debieran establecerse antes de utilizar de manera fiable la energía de baja frecuencia como pie para distinguir la sonoridad?

Lugar de articulación

Contrastaremos ahora las distribuciones de ciertas mediciones acústicas para las fricativas alveolares [s z] y palatales [š ž]. El conjunto de datos que contiene a estas cuatro fricativas divididas en alveolar *frente a* palatal, es el grupo de datos **fric_AP**

T7: Utilizando el grupo de datos **fric_AP**, examine los trazados del histograma (mediante `plot_histograms`) para cada una de las seis mediciones, y estudie su capacidad para separar las dos clases.

P4: La proporción de energía entre 2 kHz y 4 kHz y entre 2 kHz y 6 kHz (**EnergíaMFR**) es una buena medición discriminante. ¿Por qué?

T8: Genere trazados dispersos de dos dimensiones utilizando el comando `plot_clusters` para los datos `fric_AP`, empleando distintos pares de mediciones. Incluya las etiquetas del fono en el trazado y examine la influencia del contexto fonético.

P5: ¿Qué contexto fonético puede causar que las fricativas alveolares parezcan palatales (según la medición de **Energía MFR**)?

P6: Basándose en su investigación de algunas de las propiedades acústicas de las fricativas del inglés americano, junto con otras propiedades fonético-acústicas tratadas en clase, describa brevemente y justifique qué mediciones acústicas utilizaría para distinguir las ocho fricativas entre sí. Especifique el/los contextos en el/los que se utilizaría cada medición. Puede utilizar mediciones distintas a las seis incluidas en los datos de MATLAB®.

Parte II: Modelado acústico

En esta parte del trabajo, trataremos de modelar las distribuciones de las mediciones acústicas examinadas durante la primera parte del trabajo. Para el resto de la tarea, nos centraremos en intentar distinguir a las cuatro fricativas [z š f ð].

T9: Utilizando el grupo de datos de **entrenamiento** (y los índices correspondientes **train-index**), genere trazados del histograma (utilizando `plot_histograms`) para cada una de las seis mediciones, y estudie su habilidad para separar las cuatro fricativas.

T10: Genere dos trazados dispersos de dos dimensiones utilizando el comando `plot_clusters` para los datos de entrenamiento. Explore distintos pares de mediciones y examine sus habilidades para discriminar entre las diferentes clases. En estos trazados no hace falta que trace las etiquetas del fono.

P7: Basándose en sus observaciones de los trazados dispersos, ¿qué dos mediciones son las más correlacionadas? ¿Cuáles son las menos correlacionadas?

T11: Modelaremos las distribuciones de los rasgos de medición acústica utilizando distribuciones Gaussianas y distribuciones de mezcla de Gaussianas. Los modelos de Gaussianas y de mezclas de Gaussianas pueden entrenarse con los comandos **train_gaussian** y **train_mixture**. Entrene un modelo gaussiano simple de *covarianza total* para el fono [z] (grupo de datos **train_z**) utilizando el comando:

```
>> [gmean, gvar] = train_gaussian(train_z);
```

El vector de medias y la matriz de covarianza se guardan en `gmean` y `gvar`, respectivamente. Podemos obtener también el correspondiente modelo de Gaussiana de *covarianza diagonal* si diagonalizamos la matriz de covarianza (observe que el vector de medias permanece igual):

```
>> dmean = gmean;  
>> dvar = diag(diag(gvar));
```

Para entrenar un modelo de Gaussiana de covarianza diagonal *mixta* de cuatro componentes para el fono [z], utilice el comando:

```
>> [mmean, mvar, mwgt] = train_mixture(train_z,4);
```

El grupo de vectores de medias, los vectores de varianza (sólo diagonales de las matrices de covarianza) y los pesos de mezcla, se guardan en `mmean`, `mvar` y `mwgt`, respectivamente.

T12: Examinaremos ahora cómo los distintos modelos de Gaussianas satisfacen las mediciones de rasgos de los datos de entrenamiento. Para hacer esto, podemos comparar los trazados de las distribuciones de medición, con aquellos de las estimaciones de densidad de los modelos de Gaussianas, mediante los comandos `plot_histograms` y `plot_gaussian`. Por ejemplo, para ver la correspondencia del modelo de Gaussiana de covarianza total con la primera dimensión (**MFCC0**) de los datos, podemos escribir:

```
>> plot_histograms(train_z,1); hold
>> plot_gaussian(gmean,gvar,1); hold
```

El comando `hold` conmuta el esperar en el trazado actual para que otro trazado pueda superponerse, o el dibujar un nuevo trazado. Estudie cada una de las seis dimensiones para observar cuál de estas está suficientemente representada por una densidad Gaussiana. Para examinar igualmente el modelo de Gaussiana de covarianza diagonal mixta, podemos utilizar el comando `plot_mixture`.

```
>> plot_histograms(train_z,1); hold
>> plot_mixture(mmean,mvar,mwgt,1); hold
```

Examine con qué exactitud el modelo de Gaussiana de covarianza diagonal mixta, satisface cada una de las seis dimensiones.

T13: El estudio de trazados unidimensionales de los datos no permite la observación de cualquier correlación entre las distintas dimensiones. Para examinar con qué exactitud captura el modelo la información de correlación, podemos superimponer trazados dispersos de los datos de entrenamiento y "trazados de contorno" bidimensionales de los modelos de Gaussiana. Los trazados de contorno muestran líneas con la misma densidad. Para hacer esto, utilice los comandos `plot_clusters` y `plot_gauss_contour`

```
>> plot_clusters(train_z,2,3); hold;
>> plot_gauss_contour(dmean,dvar,2,3); hold;
```

Los comandos de arriba dibujarán un trazado disperso de las dimensiones segunda y tercera de los datos **train_z** con los correspondientes contornos de densidad de Gaussianas de covarianza diagonal superpuestos encima. Para examinar la correspondencia de la densidad de Gaussiana de covarianza total, utilizaremos:

```
>> plot_clusters(train_z,2,3); hold;
>> plot_gauss_contour(gmean,gvar,2,3); hold;
```

Para crear un trazado parecido utilizando la densidad de Gaussiana de covarianza diagonal mixta:

```
>> plot_clusters(train_z,2,3); hold;
>> plot_mix_contour(mmean,mvar,mwgt,2,3); hold;
```

Examine la Gaussiana de covarianza diagonal, la Gaussiana de covarianza total y los modelos de Gaussiana de covarianza diagonal mixta, sobre los distintos pares de mediciones y observe con qué exactitud satisface los datos cada modelo.

P8: En general, ¿qué modelo (Gaussiana de covarianza diagonal, Gaussiana de covarianza total o Gaussiana de covarianza diagonal mixta) parece más adecuado para modelar los datos de [z] basados en sus observaciones de los distintos trazados? ¿Qué ventajas y deficiencias presenta cada modelo?

T14: Otro modo de medir con qué exactitud captura un modelo la información de los datos de entrenamiento, consiste en computar la probabilidad o la probabilidad logarítmica, puntuación de los datos de entrenamiento cuando se prueban con el modelo entrenado. Para hallar la probabilidad logarítmica total de los datos de entrenamiento, cuando se utilizan los modelos de Gaussiana de covarianza diagonal, covarianza total y covarianza diagonal mixta, utilice los siguientes comandos:

```
>> sum (log (gaussian_probs (train_z,dmean,dvar)))
>> sum (log (gaussian_probs (train_z,gmean,gvar)))
>> sum (log (mixture_probs (train_z,mmean,mvar,mwgt)))
```

Los comandos `gaussian_probs` y `mixture_probs` computan las probabilidades de cada señal de los datos determinados mediante el modelo de Gaussiana especificado o el de Gaussiana mixta. Luego tomaremos el logaritmo de las probabilidades y las sumaremos al grupo completo de datos para obtener la probabilidad total. Esto hace suponer que las señales individuales del grupo de datos son independientes entre sí.

P9: Al examinar los valores de probabilidad de los datos de entrenamiento, ¿qué modelo esperaría ejecutar mejor cuándo clasifique los nuevos datos de prueba? ¿Bajo qué circunstancias podría una probabilidad de datos de entrenamiento alta no traducirse a un modelo exacto al clasificar nuevos datos de prueba?

T15: Los modelos de Gaussiana de covarianza diagonal suponen que las distintas dimensiones de rasgos de los vectores de datos son independientes entre sí. Como vimos en los trazados dispersos de arriba, esto no siempre es así. Esto resulta también evidente si calculamos las matrices de covarianza de los datos de entrenamiento y de prueba, utilizando el comando `cov` como se muestra a continuación:

```
>> cov(train)
>> cov(test)
```

Dése cuenta de que las entradas fuera de diagonal en la matriz son no cero. En estos casos, puede que los modelos de Gaussiana de covarianza diagonal no sean la mejor elección. Dos posibles soluciones a este problema son: 1) utilizar modelos de Gaussiana de covarianza total y/o 2) transformar los vectores de datos para que las dimensiones sean independientes. Un procedimiento para realizar esto último se conoce como *análisis de los componentes principales* o PCA. Genere un nuevo grupo de vectores de entrenamiento y prueba realizando PCA mediante el comando `pcs` (escala de los componentes principales) como se muestra:

```
>> [rtrain, rtest] = pcs(train,test);
```

Este comando lleva a cabo análisis de los componentes principales en el grupo de datos `train` y utiliza los resultados para rotar y escalar los grupos `train` y `test`, guardándolos en `rtrain` y `rtest`, respectivamente. Examine las matrices de covarianza del nuevo entrenamiento transformado y datos de prueba:

```
>> cov(rtrain)
>> cov(rtest)
```

P10: Describa la estructura de la matriz de covarianza de los datos `rtrain`. ¿Por qué no es igual la matriz de covarianza de los datos `rtest` a la de los datos `rtrain`? ¿Existe alguna razón de preferencia en el uso de PCA para transformar datos frente al simple uso de modelos de Gaussiana de covarianza total?

Parte III: Clasificación de patrones

En esta parte del trabajo, utilizaremos las técnicas de modelado que examinamos en la sección anterior para llevar a cabo la clasificación de patrones. Concretamente, examinaremos la clasificación de *máxima probabilidad* (ML) y la clasificación de la probabilidad *máxima a posteriori* (MAP) con los distintos modelos de Gaussianas. Los dos comandos pertinentes son `ml classify` y `map classify`. Estos comandos toman dos grupos de datos: un grupo de entrenamiento y uno de prueba. Éstos utilizan el grupo de entrenamiento para entrenar los parámetros del modelo de gaussiana especificado y utilizar después el modelo entrenado para llevar a cabo la clasificación en el grupo de pruebas. Para una descripción más detallada de estos comandos, introduzca `help ml_classify` y `help map_classify` en la línea de comandos de MATLAB. Observe que utilizaremos las versiones PCA rotadas y escaladas de los grupos de datos de entrenamiento (`rtrain`) y de pruebas (`rset`). Para cada una de las siguientes tareas, observe cuidadosamente la exactitud del clasificador, mientras se realizan los distintos experimentos. Trate de comprender los méritos relativos y puntos flacos de cada enfoque.

T16: Realice una clasificación de ML sobre los datos utilizando una densidad Gaussiana de covarianza diagonal por clase, escribiendo el siguiente comando:

```
>> ml_classify(rtrain,train_index,rtest,test_index,'diag');
```

Este comando entrena el modelo especificado, utilizando los datos de entrenamiento y realizando clasificación en los datos de prueba, y devuelve la estadística del funcionamiento:

Porcentaje correcto: 86.885

	dh	f	sh	z
dh	235	10	0	16
f	11	263	1	5
sh	0	1	128	5
z	9	10	68	275

El “Porcentaje correcto” es simplemente la exactitud del modelo en los datos de prueba. La matriz de abajo es la *matriz de confusión*. Las filas de la matriz de confusión contienen etiquetas correctas de fonos

de los datos, mientras que las columnas contienen etiquetas de fonos planteados como hipotéticos por el clasificador). Cada entrada $c(i, j)$ de la matriz, indica el número de vectores i clasificados como clase j . Por ejemplo, los vectores 235 [ð] fueron clasificados correctamente como [ð] pero 10 fueron mal clasificados como [z] y 16 como [ʒ]. Ninguno se confundió con [ʃ]. Repita el experimento utilizando el clasificador MAP como se muestra a continuación:

```
>> map_classify(rtrain, train_index, rtest, test_index, 'diag');
```

P11: ¿Por qué el clasificador MAP presenta mayor exactitud que el clasificador ML?

T17: Realice una clasificación MAP utilizando la función de densidad de Gaussiana de covarianza total con el siguiente comando:

```
>> map_classify(rtrain, train_index, rtest, test_index, 'full');
```

T18: Realice una clasificación MAP con densidades mixtas utilizando dos Gaussianas de covarianza diagonal por densidad mixta con el siguiente comando:

```
>> map_classify(rtrain, train_index, rtest, test_index, 'mix', 2);
```

Repita el experimento de clasificación anterior varias veces y observe cómo varía el funcionamiento a medida que se van inicializando aleatoriamente las mezclas desde los distintos puntos de salida. Modifique luego el número de Gaussianas por mezcla a 4, 8, 16, 32 y 64, y observe cómo cambia el funcionamiento

P12: ¿Qué le ocurre al funcionamiento de clasificación a medida de que aumenta el número de componentes de mezcla? ¿Por qué?

P13: ¿Qué clasificador (covarianza diagonal, covarianza total o covarianza diagonal mixta) tiene el mejor funcionamiento? ¿Satisface esto lo que usted anticipó del examen de las distribuciones y probabilidades en la sección anterior? ¿Qué razones podrían explicar los funcionamientos relativos de cada uno de los clasificadores?

T19: Al examinar el funcionamiento de un clasificador sobre tamaños de grupos de entrenamiento variables, podemos obtener una idea de cómo es el entrenamiento de los parámetros del clasificador. Cuando las estimaciones de los parámetros del clasificador comienzan a converger (con crecientes cantidades de datos de entrenamiento), se espera que el funcionamiento del clasificador con los datos de entrenamiento frente a los datos de prueba también converjan. Examine el funcionamiento del clasificador gaussiano de covarianza total, utilizando un grupo de entrenamiento que contenga sólo 77 vectores de entrenamiento (**train1**) con los siguientes comandos:

```
>> map_classify(train1, train1_index, rtest, test_index, 'full');  
>> map_classify(train1, train1_index, train1, train1_index, 'full');
```

El primer comando proporciona la exactitud del grupo de pruebas, mientras el segundo facilita la exactitud del grupo de entrenamiento (ej., pruebas en el grupo de entrenamiento). Repita el experimento anterior utilizando grupos de prueba de distintos tamaños: **train2** (152 vectores), **train3** (302 vectores), **train4** (601 vectores) y **rtrain** (1199 vectores). Observe que estos grupos de entrenamiento han sido ya adecuadamente rotados por PCA y escalados.

P14: ¿Converge el funcionamiento de un clasificador en datos de entrenamiento y prueba a medida de que aumenta el tamaño del grupo de entrenamiento? ¿Cree que los modelos gaussianos de covarianza total están suficientemente entrenados con el grupo de entrenamiento total, o son necesarios datos adicionales para facilitar estimaciones más exactas de los parámetros del modelo?

T20: Cuando sólo se encuentra disponible una cantidad finita de datos de prueba, es difícil normalmente determinar si la diferencia en el funcionamiento observado de los dos clasificadores es o no considerable desde un punto de vista estadístico. El funcionamiento de un clasificador puede verse afectado también por otros factores. Recuerde el resultado de la inicialización aleatoria en el entrenamiento de modelos de mezcla gaussianos en la tarea T18.

Para probar la significancia estadística de la diferencia en funcionamiento entre dos clasificadores distintos, evaluados sobre el mismo grupo de pruebas, se puede utilizar el test de significancia de McNemar. Para realizar esta prueba es necesario generar un vector binario para cada clasificador, especificando si cada vector de prueba es correcta o incorrectamente identificado por ese clasificador. Esto puede realizarse con los siguientes comandos:

```
>> cdiag = map_classify(rtrain,train_index,rtest,test_index,'diag');  
>> cfull = map_classify(rtrain,train_index,rtest,test_index,'full');  
>> cmix = map_classify(rtrain,train_index,rtest,test_index,'mix',4);
```

El test de McNemar para la comparación del clasificador gaussiano de covarianza diagonal con el clasificador gaussiano de covarianza total, puede realizarse con el siguiente comando,

```
>> [S, N] = mcnemar(cdiag,cfull)
```

donde S es el nivel de significancia de la prueba y N es la tabla de contingencia utilizada en el test. Repita el test de McNemar para comparar el clasificador gaussiano de covarianza total con el clasificador gaussiano de covarianza diagonal mixta, y para comparar el clasificador gaussiano de covarianza diagonal con el clasificador gaussiano de covarianza diagonal mixta. Típicamente, la diferencia en el funcionamiento de los dos clasificadores se considera estadísticamente relevante si el nivel de significancia es de .005 o menor.

P15: ¿Cuáles son los niveles de significancia de McNemar cuando se compara el clasificador gaussiano de covarianza diagonal con el de covarianza total? ¿El clasificador gaussiano de covarianza total frente al clasificador gaussiano de covarianza diagonal mixta? ¿El clasificador gaussiano de covarianza diagonal frente al clasificador gaussiano de covarianza diagonal mixta? Teniendo en cuenta estos valores, ¿cómo describiría cualitativamente el funcionamiento relativo de cada uno de los tres clasificadores? MATLAB es una marca registrada de MathWorks.