**Máster de Visión Artificial**

**Asignatura: Reconocimiento de Patrones**

**Práctica 2: Reducción de dimensionalidad**

**Orión García Gallardo**

**DNI: 48330747L**

INTRODUCCIÓN

MÉTODO DESARROLLADO

EVALUACIÓN

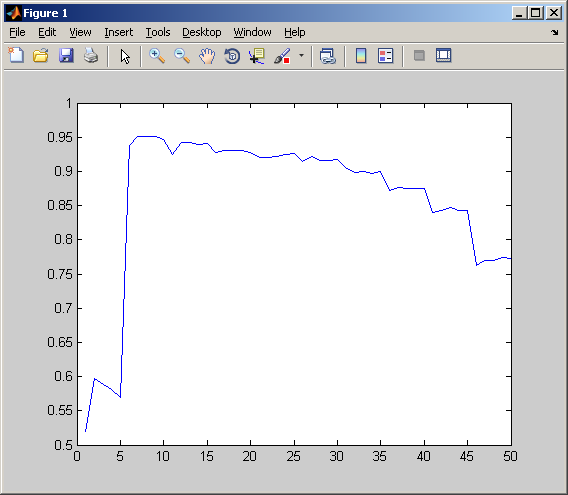
Prueba de que el LDA da malos resultados

Prueba del lda con el rendimiento

Para realizar la evaluación de los algoritmos se han hecho pruebas con las imágenes aportadas en la práctica. Dichas imágenes se dividen en 4 clases: ceda al paso, peligro, prohibición y stop. Cada clase cuenta con 111, 259, 277 y 205 imágenes respectivamente. El objetivo de la evaluación será encontrar la tupla número de características, número de vecinos más cercanos (d, k) donde el rendimiento de clasificación sea más alto.

Como primera prueba se cogen valores posibles de d en todo el espacio de características, es decir, entre 4 y 624. Se escoge 4 (el número de clases) porque el LDA no puede hacer mayor reducción de características. Para acelerar la prueba se coge un incremento de estos valores de 68 unidades. Esto significa que los valores de d serán {4, 72, 140, 208, 276, 344, 412, 480, 548, 616}. Por otro lado, los valores escogidos para k serán {1, 11, 21, 31, 41}.

El tiempo empleado en ejecutar esta prueba es de 194.632561 segundos.



0.9516 72.0000 11.0000 7.0000

0.9516 72.0000 21.0000 8.0000

0.9516 72.0000 31.0000 9.0000

0.9471 72.0000 41.0000 10.0000

0.9425 140.0000 11.0000 12.0000

0.9423 140.0000 21.0000 13.0000

0.9412 140.0000 41.0000 15.0000

0.9400 140.0000 31.0000 14.0000

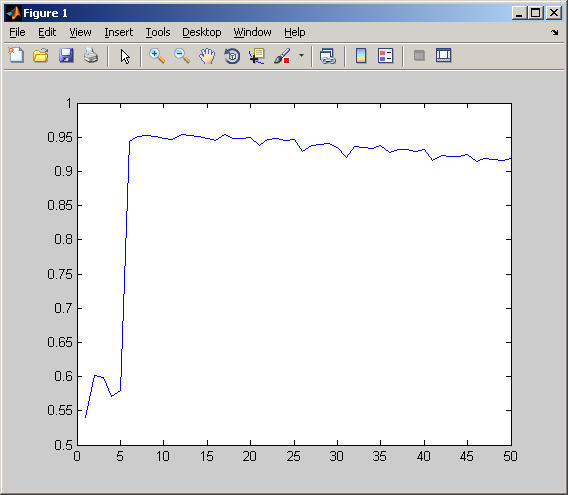
0.9384 72.0000 1.0000 6.0000

0.9307 208.0000 11.0000 17.0000

Se puede observar que el mejor rendimiento se obtiene de la mitad de la gráfica hacia la izquierda, es por ello por lo que se realiza una nueva evaluación de resultados pero cogiendo como máximo la mitad de las características, esto es hasta 312 (en matlab size(datos, 2)-1)/2).

Elapsed time is 57.777067 seconds.

%featuresEnd = (size(X, 2)-1)/2;



0.9542 106.0000 11.0000 17.0000

0.9541 72.0000 11.0000 12.0000

0.9529 72.0000 21.0000 13.0000

0.9518 38.0000 21.0000 8.0000

0.9508 38.0000 11.0000 7.0000

0.9506 38.0000 31.0000 9.0000

0.9506 72.0000 31.0000 14.0000

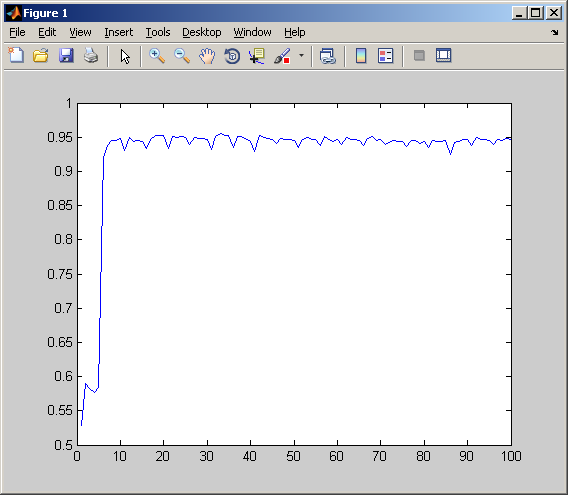
0.9496 106.0000 41.0000 20.0000

0.9484 106.0000 21.0000 18.0000

0.9484 106.0000 31.0000 19.0000

Se puede observar que el tiempo empleado es menos de la mitad, sin embargo, como en el ejemplo anterior el mejor rendimiento se vuelve a obtener en la mitad izquierda de la gráfica. La siguiente prueba se realiza con hasta 156 características (en matlab size(datos, 2)-1)/4). A su vez se aprovecha para disminuir el intervalo a la hora de escoger las características. Se elige iterar cada 8 características (en matlab floor((featuresEnd-CLASSNUMBER)/19))

Elapsed time is 45.428378 seconds.



0.9555 52.0000 21.0000 33.0000

0.9529 28.0000 31.0000 19.0000

0.9529 28.0000 41.0000 20.0000

0.9528 28.0000 21.0000 18.0000

0.9519 52.0000 31.0000 34.0000

0.9519 68.0000 11.0000 42.0000

0.9519 52.0000 41.0000 35.0000

0.9509 60.0000 21.0000 38.0000

0.9509 92.0000 11.0000 57.0000

0.9508 36.0000 31.0000 24.0000

0.9508 52.0000 11.0000 32.0000

0.9508 60.0000 11.0000 37.0000

0.9508 108.0000 21.0000 68.0000

0.9506 36.0000 11.0000 22.0000

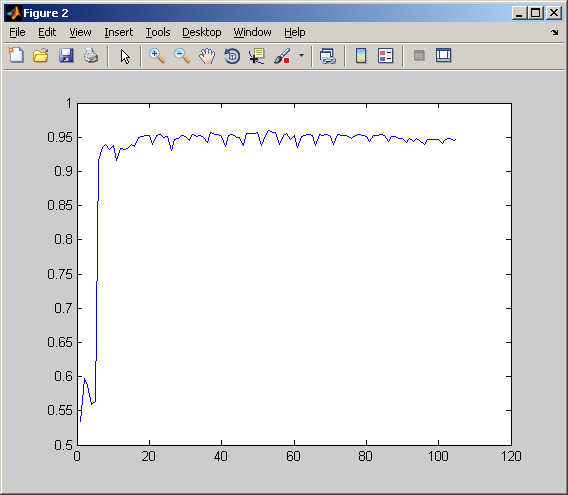
0.9496 68.0000 21.0000 43.0000

Se puede observar que el tiempo de ejecución es mejor que en el caso anterior pese a estar probando con más características. El factor determinante es que se prueba con menos dimensión de características aunque se hagan más iteraciones. Sin embargo, seguimos observando que entre los 15 mejores resultados sigue sin haber ningún valor que supere los 75% de los valores más altos de número características. Esto hace que se opte por elegir menos características, llegando al 20% de las características iniciales (o en matlab sería featuresEnd = round((size(datos, 2)-1)/5))

Elapsed time is 40.117755 seconds.

featuresMax = round((size(X, 2)-1)/5);

interval = floor((featuresMax-CLASSNUMBER)/19);



0.9592 64.0000 21.0000 53.0000

0.9567 46.0000 11.0000 37.0000

0.9567 64.0000 31.0000 54.0000

0.9567 64.0000 41.0000 55.0000

0.9566 58.0000 41.0000 50.0000

0.9555 58.0000 11.0000 47.0000

0.9555 58.0000 21.0000 48.0000

0.9554 58.0000 31.0000 49.0000

0.9554 70.0000 21.0000 58.0000

0.9544 40.0000 11.0000 32.0000

0.9544 76.0000 31.0000 64.0000

0.9544 88.0000 11.0000 72.0000

0.9544 100.0000 31.0000 84.0000

0.9542 28.0000 21.0000 23.0000

0.9542 82.0000 11.0000 67.0000

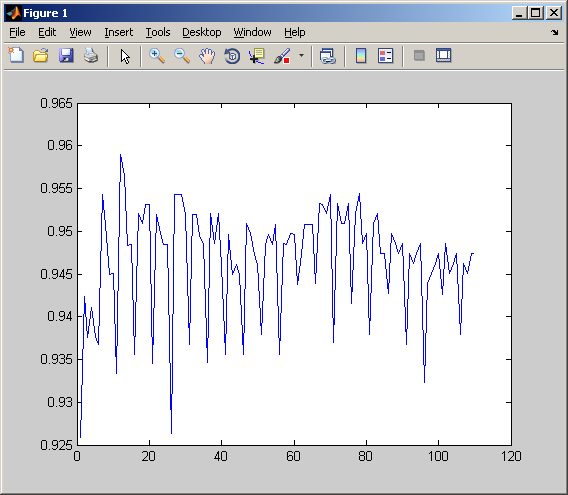
Elapsed time is 42.580749 seconds.

featuresStart = CLASSNUMBER + round(((size(X, 2)-1)/5)/10);

%featuresEnd = size(X, 2)-1;

featuresEnd = round((size(X, 2)-1)/5);

interval = floor((featuresEnd-featuresStart)/19);



0.9590 26.0000 11.0000 12.0000

0.9566 26.0000 21.0000 13.0000

0.9544 91.0000 21.0000 78.0000

0.9542 21.0000 11.0000 7.0000

0.9542 41.0000 11.0000 27.0000

0.9542 41.0000 21.0000 28.0000

0.9542 41.0000 31.0000 29.0000

0.9542 81.0000 41.0000 70.0000

0.9532 81.0000 11.0000 67.0000

0.9532 86.0000 11.0000 72.0000

0.9532 86.0000 41.0000 75.0000

0.9531 31.0000 31.0000 19.0000

0.9531 31.0000 41.0000 20.0000

0.9531 81.0000 21.0000 68.0000

Elapsed time is 43.320901 seconds.

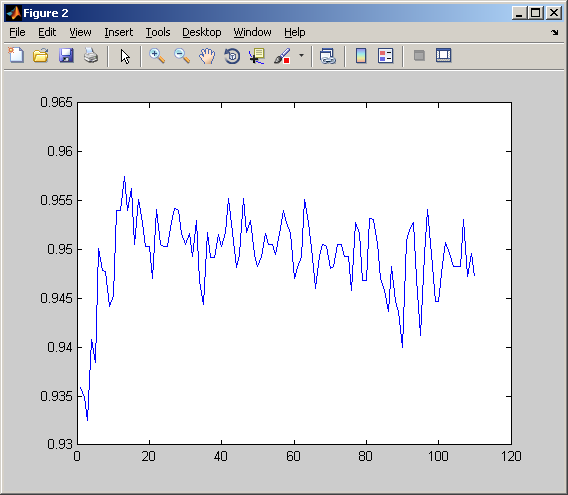
**kvalues = 5:10:50;**

featuresStart = CLASSNUMBER + round(((size(X, 2)-1)/5)/10);

%featuresEnd = size(X, 2)-1;

featuresEnd = round((size(X, 2)-1)/5);

interval = floor((featuresEnd-featuresStart)/19);



0.9575 26.0000 25.0000 13.0000

0.9563 26.0000 45.0000 15.0000

0.9553 56.0000 15.0000 42.0000

0.9553 61.0000 5.0000 46.0000

0.9551 31.0000 15.0000 17.0000

0.9551 76.0000 25.0000 63.0000

0.9542 41.0000 15.0000 27.0000

0.9541 36.0000 15.0000 22.0000

0.9541 111.0000 15.0000 97.0000

0.9540 26.0000 5.0000 11.0000

0.9540 26.0000 15.0000 12.0000

0.9540 26.0000 35.0000 14.0000

0.9540 41.0000 25.0000 28.0000

0.9540 71.0000 15.0000 57.0000

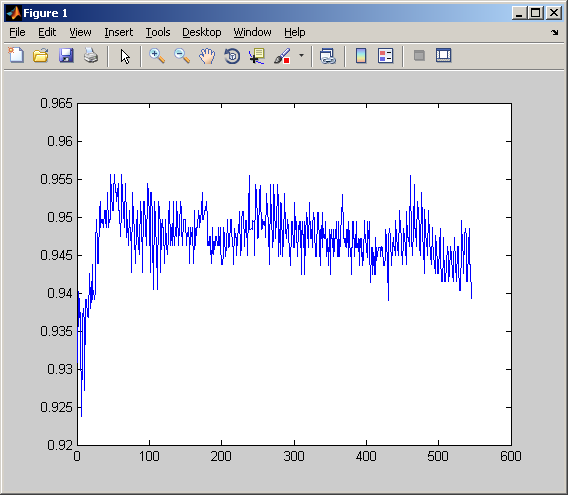
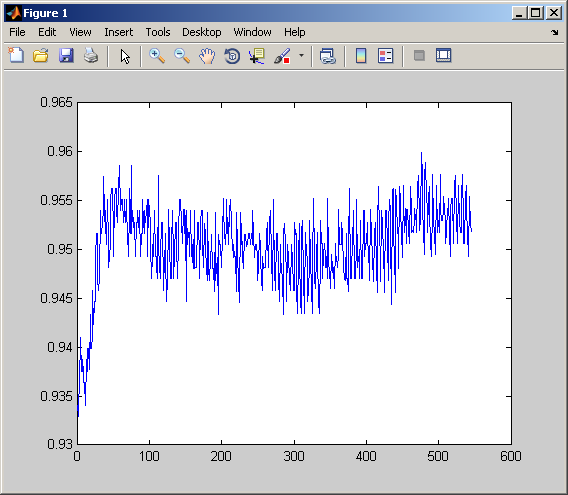
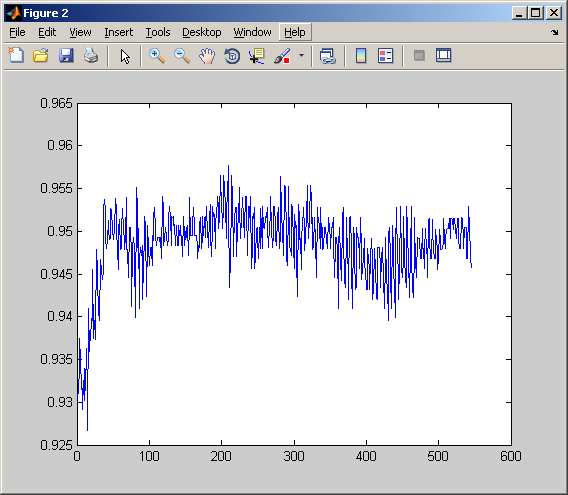
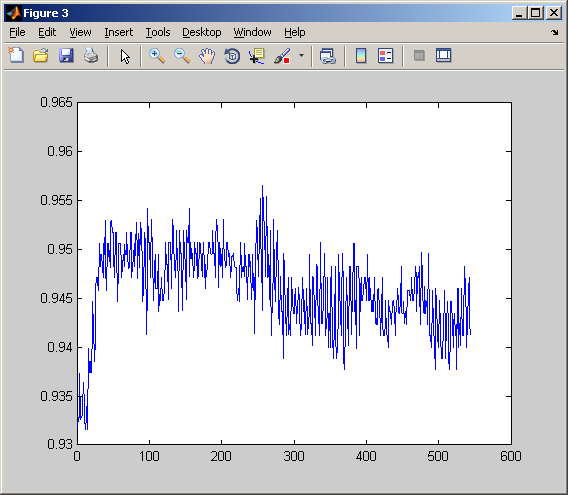
0.9532 96.0000 5.0000 81.0000

Mean elapsed time is 179,6676 seconds.

featuresStart = 16;

featuresEnd = 124;

interval = 1;

A continuación se muestra el resumen tras 4 ejecuciones:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Nº Features | K | 1º Test | 2º Test | 3º Test | 4º Test | Mean | x axis |
| 25 | 15 | 0,955542 | 0,955134 | 0,952673 | 0,952809 | 0,95404 | 47 |
| 70 | 15 | 0,954243 | 0,955134 | 0,952945 | 0,953081 | 0,953851 | 272 |
| 26 | 25 | 0,953217 | 0,956297 | 0,953836 | 0,951782 | 0,953783 | 53 |
| 25 | 25 | 0,953217 | 0,956297 | 0,952537 | 0,952945 | 0,953749 | 48 |
| 26 | 35 | 0,953217 | 0,956297 | 0,952537 | 0,951782 | 0,953458 | 54 |
| 69 | 15 | 0,954243 | 0,953972 | 0,951782 | 0,951918 | 0,952979 | 267 |
| 68 | 15 | 0,953081 | 0,951646 | 0,951646 | 0,955406 | 0,952945 | 262 |
| 27 | 35 | 0,950891 | 0,958623 | 0,951374 | 0,950619 | 0,952877 | 59 |
| 56 | 25 | 0,948565 | 0,954108 | 0,956569 | 0,950755 | 0,952499 | 203 |
| 111 | 5 | 0,953217 | 0,959921 | 0,946859 | 0,949728 | 0,952431 | 476 |

Como puede observarse tanto en las gráficas como en la tabla de resultados hay tres zonas diferenciadas con resultados bastante eficientes. La primera, la de la izquierda en las gráficas (valores del eje de las x entre 45 y 60), con resultados más estables y que se correspondería con seleccionar en el PCA entre 25 y 30 número de características. La segunda, más central en las gráficas (valores del eje de las x entre 200 y 275), con resultados algo más inestables que en la zona anterior, pero obteniendo buenas medias. Y la última zona, la de más a la derecha, (valores del eje de las x entorno al 500), con resultados muy inestables y que, por lo general, obtienen peores medias. Ejemplo de ello se puede observar en la última línea de la tabla de resultados, donde en la segunda prueba llega a obtenerse el mejor rendimiento de todas las ejecuciones (0,959921). Sin embargo, debido a los malos resultado obtenidos en las otras pruebas su media desciende hasta la décima posición.

CONCLUSIONES