

Tarea 3-Reconocimiento de Patrones

Felipe Neut

I. MOTIVACION

El reconocimiento de patrones ha adquirido una gran relevancia en numerosas aplicaciones de nuestra vida cotidiana para automatizar procesos. Es por ello que en esta tarea se adentrará a 3 aplicaciones del reconocimiento de patrones: detección de tortillas, caras, y reconocimiento de género.

II. SOLUCIÓN PROPUESTA

La solución propuesta de selección de características y clasificador utilizado difiere para cada conjunto de datos, no obstante, hubo un patrón similar de resolución, el cual se muestra a continuación.

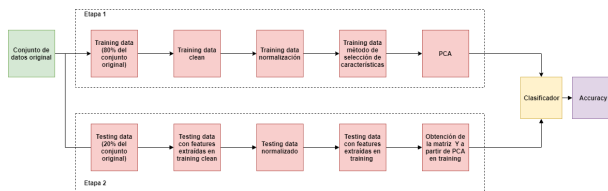


Fig.1 Solución propuesta

Es importante recalcar que se tuvo especial cuidado en las *features*, eliminando aquellas que entregaran una falsa correlación, como por ejemplo las *features* de posición.

Otro detalle a destacar es que para la selección del clasificador, siempre fue utilizado el método PCA, el cual permitía reducir la dimensión del problema. Esto entregaba la facultad de graficar mediante *Bio_plotfeatures()* como estaban siendo repartidas las clases para cada *feature* seleccionada.

Una vez dicho lo anterior, se procederá a explicar el método de selección de *features* y clasificador utilizado para cada conjunto de datos en específico.

1. Detección de caras:

→ **Training-Selección de características:** Para este caso, SFS resultó ser ineficiente, dado que se maximizaba el J de Fisher en gran cantidad con cada nueva *feature* seleccionada, resultando en la selección de casi todo el mapa de *features*. Dada la covarianza entre clases, el uso de la transformada PCA resultó ser una muy buena aproximación, producto de la propiedad de independencia lineal que entregan las columnas de Y en PCA y el no uso de la información de supervisión (a que clase pertenece cada muestra).

→ **Clasificador:** Se utilizó el clasificador QDA, el cual supone una distribución gaussiana. Se descartaron el uso de métodos de clasificación lineal como LDA y distancia mínima dado que se obtuvo que las clases se encontraban mezcladas para las *features* extraídas.

2. Tortillas:

→ **Training-Selección de características:** Se utilizó SFS con 40 *features* extraídas, las cuales entregaban una gran separabilidad de las clases para este *dataset*.

→ **Clasificador:** Se empleó el clasificador KNN con $k=1$. Dada la gran separabilidad obtenida, la línea de decisión podía ser trazada de manera más sencilla. Aún así se eligió KNN dado que es bastante robusto y tiene un bajo cómputo. No obstante, también se probó con otros clasificadores lineales como LDA, obteniendo el mismo *accuracy* (100 %).

3. Género:

→ **Training-Selección de características:** Se procedió a utilizar el método SFS, con clasificador KKN, con $k=8$. Es importante mencionar que este último método utiliza el *accuracy* obtenido por el clasificador KKN como función objetivo de la separabilidad. Una de sus desventajas es que es más costoso computacionalmente debido a que hace un *training* en cada iteración. La búsqueda a través de este método era encontrar *features* de textura y Gabor, que entregaban una mejor separabilidad de las clases.

→ **Clasificador:** Se utilizó el clasificador KNN con $k=1$, el cual permite obtener una línea de decisión con mucho más detalle, lo cual era requerido.

III. EXPERIMENTOS REALIZADOS

Según los métodos destacadas previamente, para tortillas se obtuvo un *accuracy* de 1 con un tiempo de simulación de 2.8 segundos, tomando 40 *features* de SFS y luego aplicando transformada de PCA despreciando λ mayor a 8. Por otro lado, para género se obtuvo un *accuracy* de 0.959 con un tiempo de simulación de 3.1 segundos, extrayendo 4 características de SFS con KNN con $k=8$, y luego utilizando transformada de PCA, para la cual se eligieron los 4 λ más significativos, tomando el resto de λ como despreciables. Finalmente, para el *dataset* detección de caras, dada la ineficiencia de SFS por razones explicadas previamente, utilizando exclusivamente PCA con $\lambda = 10$ se obtuvo un *accuracy* de 0.9623. Ahora, si se utiliza SFS con un gran número de extracción de *features*, específicamente 250 *features*, efectivamente se logra el *accuracy* igual a 1, no obstante, el tiempo de simulación aumentó en 51 segundos con respecto al caso utilizando exclusivamente PCA. En el código adjuntado se muestran ambos casos.

IV. CONCLUSIONES

A modo de conclusión, se utilizaron distintos métodos de selección de características y clasificadores para 3 distintos tipos de *dataset*, los cuales presentaron niveles de *accuracy* de 100 %, a excepción del *dataset* género con un 96.23 %. Se utilizaron clasificadores que entregaran una línea de decisión con más detalle cuando las clases se encontraban mezcladas para el conjunto extraído de *features*, como es el caso de KNN con $k=1$ o QDA, lo cual fue realizado para género y caras.